ADDALY · THE AI CLASSROOM

Editing an Image

Layers, masks, curves and export — the craft behind a photo that still holds up at full size.

8 modules · 78 lessons · 29 figures · 8 case studies · 74,189 words · about 12 hours

Dr Mustafa Mun

Author and editor

ABOUT THIS BOOK

Editing an Image, published by Addaly, 2026. Author and editor: **Dr Mustafa Mun**.

The manuscript was drafted with the assistance of a large language model and was edited and published under his name. That is stated plainly here rather than left to be discovered, because a reader is entitled to know how a book they are trusting was made.

This book is the printed form of the course of the same name at addaly.com/learn/image-editing. The website is the living version and is corrected first; where the two differ, the website is right.

Free to read, free to download, free to print, and free to teach from. There is no paid edition and there never will be. If you paid for this, somebody has sold you something that was given away.

Every diagram in it is drawn from data by code, not generated as a picture, so the numbers on the page are the numbers in the argument. Answers to every exercise, test and examination question are at the back, with a note on why each wrong option attracts people — those notes are the teaching, not the marking.

Contents

2. The non-destructive workshop

Build and hand over a layered document another editor can take apart — correctly ordered, grouped, masked, named and versioned — and diagnose from the stack alone why an adjustment is affecting the wrong layers or why the file has become too slow to work in

1. Every edit should still be undoable next week
2. The eraser is the tool that costs you the second revision
3. Twenty-seven blend modes, six ideas
4. The same three edits in a different order give a different picture
5. Why an adjustment stopped working when you put it in a folder
6. One file, forty mockups, one change
7. The panel where selections go to live
8. The file you will need is the one from Tuesday
9. Eleven shortcuts and a scratch disk
10. Working on a laptop that cannot hold the job
11. Case study: The yearbook, the eight-gigabyte laptop, and the Tuesday crash
12. Module test

3. Selections, cut-outs and edges

Choose and execute the right extraction method for a given edge — path, painted mask, colour range, luminosity, model or blend-mode composite — decontaminate the result so it carries no trace of its old background, and run a batch of cut-outs to a consistent standard with a sampling check that catches the failures

1. Selections are easy until you reach hair
2. Two handles and a curve: making the pen tool obey
3. Stop fighting the selection tools and paint it
4. Tolerance is a wall, fuzziness is a slope
5. Masks with no edge anywhere
6. What the one-click button is actually doing
7. Three equations, seven unknowns: why hair is hard
8. The four subjects that are not opaque
9. Four hundred cut-outs, and the twelve that are wrong
10. Case study: Four hundred and twelve brass items, and the twelve that were not
11. Module test

4. Tone and colour

Take a set of unevenly lit photographs to a consistent, defensible tonal and colour result — diagnosing casts from per-channel evidence, correcting mixed lighting locally, shaping form without banding or halos, and stating what you measured rather than what looked right on your screen

1. Brightness/Contrast is one knob for a problem with five
2. There is no correct histogram
3. Two light sources, one global slider
4. Shaping light by hand, on a grey layer
5. Why Clarity makes halos, and what Dehaze is estimating
6. Saturation is a slider that destroys detail quietly
7. Monochrome is a decision about which colours become which greys
8. A look is a mapping, and a LUT is a table of it
9. There is no correct number for skin
10. Forty photographs that have to look like one shoot
11. Case study: One hundred and eighty graduates, one corridor, and a camera that kept deciding
12. Module test

5. Repair and retouching

Repair a damaged or imperfect photograph to a professional standard using the right mechanism for each fault — separated frequencies, patch synthesis, manual reconstruction or warp — and state precisely which faults in a given file are unrecoverable and why

1. Three tools that all remove a spot, and they are not interchangeable
2. Splitting texture from tone
3. Keep what is permanent, remove what is passing
4. The product has to be the product
5. A reflection is a photograph of the light source
6. Rebuild the structure first, then the texture
7. Moving pixels, and the straight line that gives it away
8. Getting the photograph off the paper
9. Tears, mould and the cyan that faded first
10. Four losses that no tool undoes
11. Case study: The 1971 wedding photograph, and the face that is only in one other frame
12. Module test

6. Compositing: making two pictures one

Combine elements from different photographs into one image that survives inspection — matching perspective, scale, light direction and quality, edge softness, depth of field and grain — and merge multiple exposures, viewpoints or focus planes of one scene without halos, ghosting or parallax errors

7. AI inside the image editor

Use generative fill, outpainting, upscaling, denoising and colourising deliberately — predicting each one's characteristic failures from its mechanism, running free local equivalents where confidentiality or cost requires it — and state for a given image whether generated content is permissible and what has to be disclosed

8. Out of the editor: size, sharpness and the delivered file

Take a finished edit to a delivered file for any named destination — computing the pixel dimensions from the destination rather than the source, choosing a resampling filter and a colour space by mechanism, sharpening last and at output size, selecting the codec and quality by measurement rather than habit, and predicting what a platform's own re-encoding pipeline will do to the result

Editing an Image — final examination

The paper that opens the next course. Answers to everything follow it.

MODULE 1

What an image is made of

Before any tool, the file. A photograph is a grid of numbers with a colour meaning attached, and almost every mystery in editing — the soft enlargement, the banded sky, the grey fringe round a logo, the colours that will not print — is a property of that grid rather than a bug in the software. This block takes the file apart: channels, bit depth, gamma, alpha, the raw mosaic, the inside of a JPEG, gamut, your monitor, and what the lens did before any of it reached you.

> BY THE END OF THIS MODULE YOU CAN
>
> Open an unfamiliar image file and state what it can and cannot survive — its bit depth, colour space, alpha, compression history and lens geometry — and predict from those facts which edits will band, fringe, clip or fail to print, before making any of them

1 · 9 min

The photo you were sent is too small, and nothing fixes that

THE ONE THING TO KEEP

Enlarging can only average pixels you already have; AI upscaling invents new ones, which is a different thing from recovering them.

An image is a grid of numbers

Open any photograph and what you have is a rectangle of coloured squares. A 3000 by 2000 pixel photo holds six million of them. Each one is three numbers in an ordinary 8-bit file — how much red, how much green, how much blue, each from 0 to 255.

That is the whole story, and almost every beginner disaster follows from it. The file records what the camera measured. It does not record the scene. Anything the sensor never sampled is not in there, and no software can go and get it.

Why enlarging goes soft

You have a 400 by 400 logo. You need it at 2000 by 2000. The editor will do it happily and the result will look like it was printed on a wet newspaper.

Here is the mechanism. To make 2000 pixels out of 400, the software has to produce five pixels where it has one. It does that by **interpolation** — looking at the neighbouring pixels it does have and calculating a weighted average for each gap.

- **Nearest neighbour** copies the closest pixel. You get hard blocks. Useful only for pixel art and screenshots of pixel art.
- **Bilinear** averages the four surrounding pixels. Fast, mushy.

- **Bicubic** and **Lanczos** use a wider neighbourhood and a smarter weighting curve. Better, still averages.

Every one of these is a guess between two known values. A guess between two known values cannot contain a detail that sat between them in the real world and was never recorded. That is why enlargement looks soft — not because the algorithm is bad, but because there is nothing there.

Shrinking is the opposite and is nearly free. Going from 4000 pixels wide to 1000 throws away information you had a surplus of. Photos almost always survive it well.

The number that is not really in the file

People say "make it 300 dpi" and think they have said something about quality. They have not.

A JPEG has a pixel count and a metadata field saying what physical size it would like to be printed at. Changing that field changes nothing about the pixels. A 3000 by 2000 photo tagged 72 ppi and the same photo tagged 300 ppi are byte-for-byte identical in the part that matters.

What ppi tells you is arithmetic. 3000 pixels at 300 pixels per inch prints ten inches wide. The same 3000 pixels at 150 ppi prints twenty inches wide and looks fine, because a poster is read from two metres away. Billboards are routinely output at 10 to 20 ppi and nobody notices. Viewing distance is the real variable, and 300 ppi is a rule of thumb for something held in the hand.

In Photoshop this lives in **Image > Image Size**, and the checkbox that matters is *Resample*. Uncheck it and you are only changing the print arithmetic — the pixels stay put. Leave it checked and you are actually inventing or discarding pixels. Photopea, GIMP (**Image > Scale Image** versus **Image > Print Size**) and Krita all draw the same distinction, sometimes in two separate dialogues.

Where AI upscaling changes the answer, and where it does not

Model-based upscalers — Real-ESRGAN and the free desktop app **Upscayl** built on it, Topaz Gigapixel on the paid side, Adobe's Super Resolution inside Camera Raw — do something different from bicubic. They were trained on millions of pairs of small and large images, so instead of averaging they predict what a plausible high-resolution version would look like.

On some material this is remarkable. Skin, foliage, fabric, brickwork, general photographic texture: a good model produces a convincing 2x or 4x that beats any interpolation.

On other material it is dangerous, and the reason is in the word *plausible*. The model is not recovering your detail. It is generating detail consistent with the low-resolution evidence. So:

Two ways to make a 400-pixel logo 2,000 pixels wide

Interpolation: bicubic, Lanczos

- Calculates a weighted average between pixels you already have
- Cannot contain a detail that sat between two samples and was never recorded
- Fails honestly: the result is soft, and it looks soft
- Safe on a number plate, a document scan, a medical image

A trained model: Real-ESRGAN, Upscayl

- Predicts a plausible high-resolution version from millions of training pairs
- Genuinely convincing at 2 times on photographic texture
- Fails confidently: invented letterforms, regenerated faces, the same person slightly not themselves
- Changes the record, so it belongs in the delivery note

Both produce a larger file. Only one of them is honest about where the extra pixels came from. On skin, foliage and brickwork the model wins easily; on a brand mark it returns letterforms that are crisp and subtly the wrong shapes, and nothing in the interface says which case you are in.

- **Text and logos come back wrong.** The letterforms are invented. They look crisp and they are subtly the wrong shapes. A brand mark rebuilt this way will not match the real one.
- **Faces get replaced, not restored.** Face-enhancement models regenerate the features. The person comes out smoother, younger and slightly not themselves.
- **Anything used as evidence is off limits.** A number plate, a document scan, a medical or forensic image. What comes out is a guess wearing the clothes of a fact.

If the output will be presented as a record of something real, an upscaler that invents detail has changed the record, and that belongs in the labelling conversation in making things with AI.

What to do today

Ask for the original before you start. Not the WhatsApp copy, not the one pasted into a slide — the file off the camera. Ninety per cent of "can you make this sharper" is solved by getting a file that was never shrunk.

If the client only has the small logo, do not upscale it. Redraw it as vector, which has no resolution at all — that is vector and print.

And check before you accept the job: open the image, note the pixel dimensions, divide by the printed width in inches. If the answer is under about 120, say so on day one rather than on the day of the print run.

BEFORE YOU MOVE ON

A client supplies a 640 by 480 logo pulled from an old website and wants it on a two-metre banner. The designer runs it through an AI upscaler at 8x. On screen it looks sharp, but the client says the letters are subtly wrong. What happened?

A. The upscaler was run at too high a strength; a gentler setting would have preserved the original letterforms.
B. The file was exported as JPEG, so compression damaged the fine edges of the type.
C. The model had no record of the true letterforms, so it synthesised edges consistent with the blurry evidence; the fix is to redraw the logo as vector.
D. The image was left at 72 ppi, so the banner printed at low resolution; setting it to 300 ppi before upscaling would have fixed it.

2 · 9 min

Three numbers per pixel, and why green counts most

THE ONE THING TO KEEP

A colour pixel is three independent brightness values whose contributions to perceived lightness are roughly 21/71/7 per cent, so the green channel is your cleanest and averaging the three is never a correct greyscale.

Every colour image is three grey images

Open the Channels panel in Photoshop, Photopea or GIMP and switch off two of them. What is left is a greyscale picture. That is the whole structure of a colour file: three separate brightness maps, one for red, one for green, one for blue, stacked and displayed together.

An 8-bit file gives each channel 256 levels, 0 to 255. Three channels of 256 levels is 16,777,216 possible combinations, which is where the phrase "16.7 million colours" comes from. A hex code is just those three numbers in base 16: `#FF0000` is red 255, green 0, blue 0. `#808080` is 128 in all three, which is why equal values always give you a neutral grey.

This is **additive** colour. It describes light being emitted, so all three at maximum gives white and all three at zero gives black. Ink works the other way round — it subtracts light from white paper — which is why print uses CMYK and why some screen colours cannot be printed at all. That is later in this module.

The channels are not equal, and your eye is why

Here is the fact that changes how you work. Brightness is not the average of the three numbers. The standard weighting used by almost every piece of software, from JPEG encoders to video, is:

```
luminance = 0.2126 × R + 0.7152 × G + 0.0722 × B
```

Green carries roughly 71 per cent of the perceived brightness, red 21 per cent, blue 7 per cent. This is not a convention somebody picked. The human retina has far more cones sensitive to the middle of the spectrum than to the ends, so a pure green patch genuinely looks much brighter to you than a pure blue patch of the same numeric value.

Two practical consequences follow immediately.

The green channel is usually your cleanest one. Camera sensors have twice as many green photosites as red or blue, so the green channel has the least noise and the most real detail. When you need a channel to build a mask from, start by looking at green. The blue channel is usually the noisiest, which is also why blue skies band first.

Desaturating by averaging is wrong. Set a Hue/Saturation layer to −100 and some tools average the channels. A red jumper and a blue jumper that look obviously different in colour come out as near-identical greys, and the picture goes flat. Use a proper Black and White adjustment, which lets you set the weight per channel. There is a whole lesson on this later.

HSB is a steering wheel, not a storage format

The colour picker offers you Hue, Saturation and Brightness. Nothing in the file is stored that way. HSB is a re-parameterisation of the same three numbers into controls that match how people talk: hue as an angle round a wheel, saturation as distance from grey, brightness as overall level.

It is genuinely easier to work with — "make it slightly less orange" is one slider in HSB and three in RGB — but it hides a trap. HSB brightness is not the luminance formula above. Pure yellow at HSB brightness 100 and pure blue at HSB brightness 100 differ by about a factor of ten in actual light. Move a hue slider and the perceived lightness moves with it, which is why a hue shift on a logo often needs a lightness correction afterwards.

Look at your channels once and you will keep doing it

A two-minute habit worth building. In any editor, cycle the channels on a portrait:

- **Red** — skin goes pale and smooth, because skin reflects a lot of red. Blemishes almost vanish. This is why frequency separation and skin work often lean on red.
- **Green** — the most "normal" looking, and the sharpest.
- **Blue** — skin goes blotchy and dark, hair separates hard from a pale background, noise is obvious.

That blue channel, ugly as it is, is often the fastest route to a hair mask. In GIMP: **Colours > Components > Decompose** to see all three as separate layers. Photopea and Photoshop both have the Channels panel with the same keyboard shortcuts.

The honest limit: three numbers are a model of you, not of light

Real light arriving at a surface is a continuous spectrum — energy at every wavelength. Three numbers cannot record a spectrum. They record how strongly that spectrum would excite three particular kinds of receptor.

The consequence has a name: **metamerism**. Two objects with completely different spectra can produce identical RGB values under one light and visibly different ones under another. This is the shirt and trousers that match under shop fluorescents and clash in daylight, and it is why a printed swatch can match a screen under one lamp and not another.

No amount of editing fixes it, because the information was never in the file. It is also why print work is checked under a standardised viewing light rather than whatever bulb is in the room, and why a client who approves a colour on a phone in a café has not really approved anything.

BEFORE YOU MOVE ON

You convert a photograph of a red flower against green leaves to black and white by dragging Saturation to −100, and the flower and leaves come out as nearly the same grey. What has actually happened?

A. The saturation slider damaged the file, and the detail can be recovered by raising saturation again before flattening.
B. Red and green happened to sit at similar numeric values, so any greyscale method would have merged them.
C. The conversion collapsed hue without weighting the channels by how bright each one looks, so two colours the eye separates easily landed on the same value.
D. The image was in the wrong colour space, and converting to Adobe RGB first would have preserved the separation.

3 · 9 min

Why the sky turns into stripes

THE ONE THING TO KEEP

Banding is empty space in the histogram — a stretch redistributes the levels you have rather than creating new ones — so do tonal work in 16-bit, dither what remains, and remember that converting a JPEG upward adds no levels at all.

Two hundred and fifty-six steps, and where they run out

An 8-bit channel has 256 possible values. In a photograph of a market, a street, a face, that is plenty — the tones jump around from pixel to pixel anyway and you will never see the steps.

A smooth gradient is different. Take a clear sky that runs from value 150 at the horizon to value 180 at the top of the frame across 1,200 pixels of image height. That is 30 available levels spread over 1,200 pixels, so each level occupies a band 40 pixels tall. The human eye can reliably detect a one per cent difference in luminance in a smooth area, and one level out of 256 near mid-grey is about half a per cent — close enough that under the right conditions you see the edges. You get **banding**: visible stripes where the software had no value in between.

Now stretch that sky. A Curves move that expands those 30 levels to fill 90 levels does not create 90 distinct values. It maps the 30 you had onto 90 slots, leaving 60 of them empty. The bands get three times taller and three times more contrasty. Open the histogram afterwards and you will see it: a comb, with white vertical gaps where no pixels remain.

Where it bites in practice

- **Skies**, especially blue ones, because the blue channel is the noisiest and gets the heaviest smoothing in JPEG.
- **Studio backdrops and seamless white**, which are pure gradient by design.
- **Dark scenes**. Because of gamma encoding, the shadow end of an 8-bit file has fewer code values per unit of actual light than the highlights do, so lifting shadows bands faster than pulling down highlights.
- **Flat colour behind a logo**, where a subtle gradient in a brand background turns into terraces on a printed banner.
- **Anything shown on a big screen**, because bands are a spatial artefact and a projector makes each one a hand's width across.

What 16-bit actually buys you

Switch to 16 bits per channel — **Image > Mode > 16 Bits/Channel** in Photoshop and Photopea, **Image > Precision** in GIMP, and Krita offers 16-bit integer and 16- and 32-bit float — and each channel gets 65,536 levels instead of 256. That is 256 times the room. Any stretch you apply still leaves gaps, but the gaps are far smaller than one 8-bit step, so when you convert back down at the end the result is smooth.

Photoshop's "16-bit" is actually 15 bits plus one, giving 32,769 levels rather than 65,536. It is a historic implementation choice, it makes no practical difference, and it confuses people who check the maths. GIMP and Krita use the full range.

The costs are real and worth naming:

- **The file doubles.** A 24-megapixel 16-bit TIFF is around 140 MB. A 30-layer document at that depth will exhaust a laptop with 8 GB of memory.
- **Some filters and plugins are 8-bit only.** Older GIMP plugins and a few Photoshop filters grey out in 16-bit.
- **Photopea's 16-bit support is partial.** It will open and convert, but several adjustments internally round to 8 bits. For heavy tonal work on a gradient, this is one of the places where the free browser tool genuinely falls short and GIMP or Krita is the better free choice.

Dither: noise as a repair

There is a second defence, and it looks like the opposite of quality. Adding a small amount of random noise breaks the boundary between two bands into a scatter of pixels from each, and your eye averages them back into a smooth transition. This is **dithering**, and it is why film grain hides banding that a clean digital gradient shows.

Practically: Photoshop's Gradient tool has a **Dither** checkbox — leave it on, always. GIMP's gradient tool has the same option. When you flatten a 16-bit file to 8-bit for delivery and still see a band in the sky, add a noise layer at 1 to 2 per cent, set to Overlay, masked to the sky. One to two per cent is enough. It will survive JPEG compression better than the band will.

The limit nobody mentions

Converting an 8-bit file to 16-bit does not recover anything. If you open a JPEG and switch it to 16 bits per channel, the file now has 65,536 available levels and still contains only 256 distinct ones. The histogram is the same comb it always was, just drawn on a wider axis.

What the conversion does is stop you adding *more* damage: further edits now round into the finer grid instead of piling up on the coarse one. That is worth doing before a big tonal move on a JPEG, and it is not the same as having shot 14-bit raw. The only way to have the levels is to have captured them.

One more honest note: the final delivered file is almost always 8-bit, because JPEG, WebP and the web are 8-bit. Working in 16 is about protecting the *process*, not the output.

BEFORE YOU MOVE ON

A retoucher opens an 8-bit JPEG, immediately converts it to 16 bits per channel, and then applies a strong Curves stretch to a flat sky. Compared with doing the same stretch in 8-bit, what is the realistic outcome?

A. Identical banding, because a JPEG can never be improved and bit depth affects only file size.
B. Noticeably less banding, because 16-bit conversion interpolates intermediate values across the gaps between the original 256 levels.
C. The same gaps between the original tones, but no additional rounding damage from the Curves move and anything done after it.
D. Worse banding, since spreading 256 values across 65,536 slots makes each gap larger and therefore more visible.

4 · 10 min

Mid-grey is not half the light

THE ONE THING TO KEEP

Stored pixel values are spaced to match your eye, not the light, so any operation that averages or adds pixels — resizing, blurring, transparency, compositing — is arithmetically wrong unless it is done in linear light.

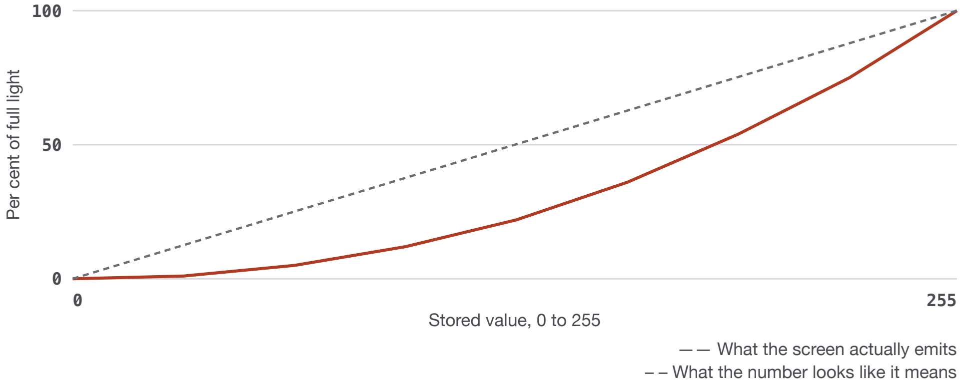
The number 128 is a lie you need

Take a pixel at RGB 255 and a pixel at RGB 128. The second one looks roughly half as bright. It is emitting about **22 per cent** of the light of the first.

This is not an error. Image files are stored with a **transfer function** applied — in sRGB, close to raising the value to the power 2.2 on the way to the display and the inverse on the way in. The reason is efficiency. Human vision is roughly logarithmic: you can tell 1 unit of light from 2 units easily, and 100 from 101 not at all. If 8-bit files stored light proportionally, almost all 256 codes would be spent on bright tones nobody can distinguish, and the shadows — where the eye is most sensitive to steps — would band horribly. Encoding perceptually spends the codes where your eye can use them.

So the numbers in an ordinary image file are **perceptually spaced**, not **physically proportional**. That single fact explains a family of bugs that otherwise look like software defects.

What a stored 8-bit value is actually emitting



Mid-grey, 128, looks about half as bright and emits about 22 per cent of the light. That gap is why averaging two stored numbers is not averaging light: a checkerboard of 0 and 255 reduced by half should come back at about 188, and software that averages the encoded values hands you 128.

Where it goes visibly wrong

Resizing. Take a fine checkerboard of pure white (255) and pure black (0) and scale it down until each new pixel covers one white and one black square. Physically, half the light is gone, so the correct answer is about 187 in sRGB. Software that averages the stored numbers gives you 128. The shrunken checkerboard comes out noticeably darker than the original, and a photograph of a distant railing or a striped shirt darkens the same way.

Blur and glow. Blurring a bright highlight across a dark background in gamma space produces a dim, grey smear rather than a spreading glow. Compositing a bright light source over a night scene is the case where this is impossible to ignore.

Alpha blending and semi-transparent edges. A 50 per cent opaque white over black should read as 187, not 128. Anti-aliased white text over a dark background, blended in gamma space, looks thinner and weaker than it should. This is the mechanism behind text that looks fine as black-on-white and spindly as white-on-black.

Saturated colour fringes. Scaling down a red-and-green pattern in gamma space can produce a dark muddy band along the boundary rather than a clean yellow-ish mix.

Doing it correctly

The fix is to convert to **linear light**, do the arithmetic, convert back. Most tools can:

- **Photoshop**: *Edit* > *Colour Settings* > *More Options* has **Blend RGB Colours Using Gamma 1.00**. It is off by default for a good reason — see the limit below.
- **GIMP 2.10 and later** works internally in 32-bit linear light for many operations and lets you choose per-operation in several dialogues. This is one of the genuine technical advantages of the free tool.
- **Krita** can set a document to 32-bit float linear, which is the standard for anyone compositing with rendered 3D.
- **ImageMagick** on the command line: `magick in.png -colorspace RGB -resize 25% -colorspace sRGB out.png`. Here `RGB` means linear and `sRGB` means encoded, which is confusing and correct.
- **Raw developers** — darktable, RawTherapee — work in linear scene data for the early part of the pipeline by design, which is one of the reasons starting from raw gives cleaner results.

When to care, and when not to

For an ordinary photograph of an ordinary scene, gamma-space resizing is usually fine. Real photographs are full of mid-tones with soft transitions, and the error is small enough to be invisible. This is why the industry got away with it for twenty years.

Care when you have **extreme brightness ranges next to each other** — a lamp at night, a window in a dark room, stars, specular highlights on chrome. Care with **graphics**: logos, flat saturated colours, thin lines, text, anything with a hard black-white boundary being scaled. Care when **compositing** semi-transparent elements, smoke, glass or rendered output. Care with **HDR and exposure blending**, where you are literally adding light values together.

The honest cost of switching it on

Turning on linear blending in Photoshop changes the result of every blend mode and every partially transparent layer in every file you open, including files a colleague made with it off. A document that looked right yesterday will look different today, and a layered file exchanged between two people with different settings will not match. That is why Adobe left it off: correctness was traded for compatibility with a decade of existing documents, deliberately.

So the sensible practice is not to flip a global switch. It is to know which operations are sensitive, handle those explicitly — a linear-light resize in ImageMagick, a 32-bit float document for a composite with real light in it — and leave the rest alone.

One genuine unresolved corner: not every operation has an obviously "correct" space. Sharpening in linear light exaggerates highlights unpleasantly, and most photographers prefer it done on encoded values even though the arithmetic is arguably wrong. "Physically correct" and "looks right" are not always the same target, and where they differ, photography has generally chosen the second.

BEFORE YOU MOVE ON

A designer scales a logo made of thin pure-black lines on pure white down to a quarter size, and the lines come out heavier and darker than expected. What is the mechanism?

A. The resampling filter used too wide a kernel, and switching from bicubic to bilinear would give correctly weighted lines.

B. Averaging perceptually encoded values understates the light in the mix, so a half-black, half-white pixel lands near 128 instead of the roughly 187 that half the light corresponds to.

C. Black lines gain weight at every reduction because black is stored as zero and zero dominates any average.

D. The file was in the wrong colour profile, so the black point was mapped below zero during the resize.

5 · 9 min

Transparency is a number, and it carries a colour with it

THE ONE THING TO KEEP

Alpha is a single coverage number per pixel that leaves the pixel's colour untouched, so transparency problems split into three separate causes — the wrong premultiply convention, contaminated edge colour, and formats that only store on-or-off — each with its own fix.

The fourth grey image

Add transparency to a file and you add a fourth channel beside red, green and blue. It holds one value per pixel, 0 to 255: 0 means this pixel shows nothing, 255 means it shows fully, and everything between is partial.

The checkerboard your editor draws behind a transparent area is not in the file. It is a display convention so you can tell transparent from white. Beginners export a PNG expecting the checkerboard to be there; it is not, because it never was.

The important thing about a partially transparent pixel is that it still has a colour. A pixel at alpha 128 is half-showing *some specific RGB value*. Which value that is turns out to be the source of most transparency problems.

Straight and premultiplied, and the fringe that tells you which

There are two conventions for what the RGB numbers mean when alpha is partial.

Straight (unassociated) alpha: RGB holds the pixel's true colour, alpha holds coverage separately. PNG works this way, and so do most editors.

Premultiplied (associated) alpha: RGB has already been multiplied by alpha. A half-transparent pure red is stored as (128, 0, 0, 128) rather than

(255, 0, 0, 128). Many video codecs, rendering engines and compositing applications use this because it makes blending a cheaper operation.

Mix them up and the symptom is immediate and diagnosable. Treat premultiplied data as straight and the edges of everything go **dark**, because the values were already reduced and you reduce them again. Treat straight data as premultiplied and the edges go **bright** or washed out. If a logo handed to you from a motion designer has a black halo on every edge, that is the bug, and the fix is a checkbox in the import dialogue rather than a mask.

The grey halo that is not a premultiply bug

There is a second, much commoner fringe, and it has a different cause. Cut a subject out of a photograph and the edge pixels are genuinely mixtures — the sensor averaged subject and background into one value. Those mixed pixels get partial alpha, and they keep the colour they had, which includes the old background.

Put that cut-out on white and you see a grey rim. Put a subject cut from a green wall onto a white page and you see a green rim. The alpha is correct. The **colour** underneath it is wrong, and no amount of contracting or feathering fixes it, because contracting just eats the edge and feathering smears the contamination further. Colour decontamination — rewriting those partial pixels' RGB to match the subject while keeping their alpha — is the only control that addresses the cause, and it gets a full lesson later in the course.

What each format can actually carry

- **PNG-32** — 8-bit alpha, 256 levels. The dependable choice for a logo with soft edges.
- **PNG-8** — one index in the palette is "transparent" and nothing else. That is 1-bit alpha: fully on or fully off. Soft edges become jagged. This is why a logo exported as a small PNG suddenly has staircase edges; it was saved as PNG-8 to make the file smaller.
- **GIF** — the same 1-bit limitation, for the same reason.

- **WebP and AVIF** — full 8-bit alpha, and considerably smaller than PNG for the same image. For web delivery of a cut-out with soft edges, WebP is usually the right answer now.
- **TIFF** — full alpha, and also supports a **clipping path**, which is a vector outline stored alongside the pixels. Print workflows often want the path rather than the alpha, because a printer's layout software uses it to flow text round the shape.
- **JPEG** — none. There is no such thing as a transparent JPEG. When a client's logo arrives sitting on a white box, this is why. The only fix is to obtain a PNG, SVG or the original artwork.
- **SVG and other vector formats** — no pixels, so no alpha channel, but full opacity control on shapes. For a logo this is always the better delivery.

Matting: telling the file which background it is going to

If you already know the final background colour, you can bake it in. Export dialogues call this **matte**. Setting the matte to the page's background makes the semi-transparent edge pixels blend towards that colour, so even a 1-bit PNG-8 or GIF looks acceptable against that one background — and visibly wrong against any other.

This is a good trick for a fixed-colour email header and a bad habit for a logo you will reuse. Deliver the untouched straight-alpha version as the master and matted copies as extras, clearly named.

Free tools throughout: GIMP has **Layer > Transparency > Colour to Alpha** and full alpha support, Photopea exports PNG-32 and WebP with alpha, and `magick in.png -background white -alpha remove out.jpg` does a matte from the command line in one line.

The limit built into a single number

Alpha is one value per pixel. It says how much of the pixel gets through, and it cannot say *which wavelengths* get through. Real coloured glass, smoke, tinted plastic and water do not attenuate all three channels equally — red glass passes red and blocks green and blue.

So a cut-out of a red wine bottle with a correct alpha channel, placed on a blue background, is wrong: the transmitted colour behind the glass should be filtered, and alpha cannot filter. The professional answer is not a better mask; it is to composite the transmissive parts with blend modes — Multiply for what absorbs light, Screen or Add for what emits it — so the background actually passes through the tint. Cut-outs of glass, flame, smoke and hair-thin highlights are a compositing problem dressed up as a selection problem, which is why they take so long.

BEFORE YOU MOVE ON

A cut-out of a red glass bottle with a clean, accurate alpha channel is placed over a blue background. The glass still looks as if it is holding the original studio background behind it rather than the blue. Why can a better mask not fix this?

A. The alpha channel is 1-bit, so partial transparency through the glass was never recorded.

B. Alpha controls how much of the pixel shows but not what colour passes through it, so tinted transmission has to be composited with a blend mode instead.

C. The file was premultiplied and is being read as straight, which is why the interior reads as the old background.

D. The mask needs decontaminating, because the glass pixels still carry colour from the studio background they were shot against.

6 · 10 min

Two thirds of the colour in your photo was calculated

THE ONE THING TO KEEP

A raw file is linear sensor data before demosaic, white balance, tone curve and 8-bit conversion, which is why white balance costs nothing and a stop of highlights can be rebuilt from the channels that did not clip.

One colour per photosite

A camera sensor measures light, not colour. Each photosite counts photons and reports a single number. To get colour, manufacturers lay a **colour filter array** over the sensor — the Bayer pattern, invented at Kodak in 1976 — so each photosite sees only red, only green or only blue.

The pattern is half green, a quarter red, a quarter blue, in a repeating two-by-two tile: green and red on one row, blue and green on the next. Twice as many green sites, for the same reason green dominated the luminance formula two lessons ago: that is where your eye gets most of its detail.

So a 24-megapixel camera captures 24 million single-colour samples. It does not capture 24 million red-green-blue triples. Every pixel in your finished file has one measured channel and two that were **calculated from the neighbours** — a process called demosaicing. Two thirds of the colour data in every digital photograph is an interpolation.

This is not a scandal, and you would never notice it on a normal subject. You notice it on fine repeating detail: a distant brick wall, a tweed jacket, a window screen, the weave in a shirt. When the detail is near the sampling frequency of the array, demosaicing produces **moiré** — coloured wave patterns that were never in the scene — and occasionally false colour speckle along high-contrast edges.

The algorithms differ and the free tools let you choose, which most photographers never realise. RawTherapee offers AMaZE, DCB, LMMSE, IGV and others; darktable offers PPG, VNG4, RCD and LMMSE. AMaZE is usually the best on detail; LMMSE is noticeably better on very noisy high-ISO files; VNG4 suppresses maze artefacts in flat areas. Switching the algorithm on a difficult file is a free improvement available to nobody shooting JPEG.

What the camera throws away when it makes a JPEG

Shoot JPEG and the camera does all of the following, permanently, before you ever see the file:

1. Demosaics, with an algorithm you cannot change.
2. Applies white balance — bakes it in.
3. Applies a tone curve, usually a contrasty one.
4. Applies noise reduction and sharpening at settings you did not choose.
5. Converts from 12 or 14 bits per channel down to 8.
6. Compresses with JPEG's lossy encoder.

A raw file is the sensor's numbers before all six. It is typically 12 or 14 bits and, crucially, **linear** — proportional to the light, with no transfer function applied yet.

The two things raw genuinely gives you

White balance for free. In a raw file, white balance is still a set of multipliers waiting to be chosen. Shot under a tube light and got a green cast? Change one number in the developer and it is gone, with no cost at all. Fix the same cast in an 8-bit JPEG and you are stretching one channel against the others inside 256 levels, which is exactly the banding problem from two lessons ago.

Highlight headroom. Here the mechanism is worth knowing precisely. The three channels clip at different points, because white balance multipliers differ — under tungsten light the blue channel is multiplied up hard, under daylight the red is. When a bright sky is slightly overexposed, the green channel may be at maximum while red and blue still hold real values. A raw developer can reconstruct the clipped channel from the two that survived and recover the tone and colour. That is where the familiar "one to two stops of highlight recovery" comes from, and it is genuine.

There is also shadow room. 14 bits linear means 16,384 levels, distributed proportionally to light, so the shadows carry far more distinct values than an 8-bit gamma-encoded file does after it has been squeezed.

The costs, stated plainly

- **Size.** A 24 MP raw is 25 to 35 MB against 6 MB for a JPEG. A wedding at 2,000 frames is 60 GB rather than 12.
- **Time.** Every raw needs developing. For 40 product shots that is fine; for a school sports day it is an evening.
- **Format churn.** Raw formats are proprietary and differ per camera — CR3, NEF, ARW, RAF, plus phone DNG. New camera models are not supported by free developers until a release ships with the right decoding data, which can be months. Adobe's free **DNG Converter** is the standard workaround: convert to DNG, and darktable and RawTherapee will open it.
- **Nothing is "finished".** A raw file always looks flatter than the camera's JPEG on first open, because the camera's contrast and saturation were not applied. Beginners conclude raw is worse. It is not finished, which is the point.

The free path is strong here. darktable and RawTherapee are both mature, both free, both run on a modest laptop, and both do things Lightroom does not — RawTherapee's demosaic choice and darktable's profiled denoise are genuinely best-in-class. ART is a friendlier fork of RawTherapee for people who find it dense. On a phone: most mid-range Android models shoot DNG in Pro mode and iPhone Pro models shoot ProRAW, and Lightroom mobile's free tier develops both.

What raw does not do

Raw fixes exposure, white balance and a stop or so of highlights. It does not fix focus, it does not fix motion blur, and it does not fix a highlight where **all three** channels are at maximum — there is nothing left to reconstruct from, and what the developer paints in there is a flat guess. A blown sky is a blown sky in any format.

It also does not rescue a composition. The commonest mistake among people who have just switched is shooting carelessly on the assumption that raw covers it. It buys you latitude, not a different photograph.

BEFORE YOU MOVE ON

A photograph of a bright overcast sky is slightly overexposed. In the raw file, the green channel is at maximum across the sky but red and blue are not. Why can a raw developer recover tone there that a JPEG of the same frame cannot?

A. Because raw files store a backup copy of the highlight data in a separate low-gain exposure written by the sensor.

B. Because raw files are 14-bit, and a 14-bit maximum is a higher light level than an 8-bit maximum.

C. Because the two channels that did not clip still hold real values, so the clipped one can be reconstructed from their ratio before white balance is baked in.

D. Because the JPEG engine applies noise reduction, which destroys the highlight detail that would otherwise have survived the conversion.

7 · 9 min

Eight by eight, and what the encoder threw away

THE ONE THING TO KEEP

JPEG loses data at exactly one step — rounding the 8×8 transform coefficients — so blocking, ringing and colour fringes each map to a specific stage of the encoder and tell you whether to change the quality, the subsampling or the format.

The encoder's four steps

JPEG has been the default photographic format since 1992 and the reason it works is worth twenty minutes of your life, because every artefact you will ever have to argue with a client about comes from one of four steps.

1. Convert to brightness and two colour differences. RGB becomes Y (luminance) plus Cb and Cr (blue-difference and red-difference). Your eye is far more sensitive to brightness detail than colour detail, and this split lets the encoder treat them differently.

2. Throw away colour resolution. In 4:2:0 subsampling — the near-universal default — the Cb and Cr channels are stored at half resolution in each direction, so one colour sample covers a 2×2 block of pixels. That is three quarters of the colour data discarded immediately, and on a face or a landscape you genuinely cannot see it. On a thin red line on white you can, which is the next section.

3. Split into 8×8 blocks and transform each one. Each block goes through a discrete cosine transform, which re-describes those 64 values as 64 coefficients: one average level plus 63 patterns of increasing fineness. Nothing is lost yet; this is a lossless re-description.

4. Quantise. Each coefficient is divided by a number from a quantisation table and rounded. The fine-detail coefficients get divided by large numbers, so most of them round to zero. **This is the lossy step, and it is the only one.** The quality slider does nothing except scale that table — quality 90 uses small divisors, quality 40 uses large ones.

The zeros then compress to almost nothing, which is where the file size comes from.

Reading an artefact backwards

Knowing the four steps means you can look at a bad file and say which step caused it.

Four steps, and the artefact each one leaves behind

Split brightness from colour
RGB becomes Y, Cb and Cr. Nothing is lost here; the split exists so the next step can treat them differently.

→

Store colour at half resolution
4:2:0 subsampling discards three quarters of the colour data. Invisible on a face. This is where red type on white gets its soft dirty fringe, and the quality slider is downstream of it.

→

Transform each 8 by 8 block
Sixty-four values become sixty-four coefficients: one average level and 63 patterns of increasing fineness. Still lossless — a re-description, not a reduction.

→

Quantise, the only lossy step
Divide each coefficient and round. Too coarse and only the block average survives, which is blocking. Truncating the sum at a hard edge leaves overshoot, which is the ringing around black text.

→

Open, edit, save again
The decoded values no longer sit on the new grid, so every round rounds again. After thirty the edges crawl and the flat areas terrace, and a rotate in an ordinary editor costs a whole generation.

Read an artefact backwards and it tells you which control to reach for. Blocking in a sky is the quality slider. A dirty fringe on red type is the subsampling, and raising the quality cannot touch it because that step happens earlier. Both are reasons the file should have been a PNG or a WebP.

Blocking — visible 8×8 squares, usually in sky or shadow. Quantisation was so coarse that only the block's average level survived. Cause: quality set too low, or an image re-saved many times.

Ringing, also called mosquito noise — a shimmer of light and dark just outside a hard edge, most obvious around black text on white. The encoder is describing a step edge as a sum of smooth cosine waves; truncating that sum leaves overshoot on either side. This is a mathematical property of the transform, not a bug, and it is why JPEG is wrong for screenshots, diagrams and text.

Colour bleed and dirty fringes on saturated text — step 2. Red type on white is a hard edge in a channel that was stored at half resolution. Raising the quality slider does not fix it, because subsampling happens before quantisation. Some encoders let you force 4:4:4 (no subsampling); Photoshop does it automatically at quality 7 and above in *Save for Web*, and `cjpeg -sample 1x1` does it explicitly. Or just use PNG or WebP, which is the real answer.

Generation loss — open a JPEG, make one small edit, save at quality 85, repeat. After 30 rounds the edges crawl and the flat areas terrace. What happens is that each decode lands on values that do not sit neatly on the new quantisation grid, so each re-encode rounds again. It does eventually converge to a stable state, but well below where it started. Re-saving *without any edit* in a tool that re-encodes still loses a little each time; re-saving with identical settings converges faster but never recovers.

Things you can do losslessly

Not every JPEG operation has to re-encode. `jpegtran` does several transformations by rearranging the coefficient blocks directly:

```
jpegtran -rotate 90 -perfect -copy all in.jpg > out.jpg
jpegtran -crop 800x600+16+16 in.jpg > cropped.jpg
```

Rotation is exact if the dimensions are multiples of 8 or 16, which `-perfect` enforces. Cropping is lossless only on block boundaries, so the offsets must be multiples of 8. Most phone gallery apps that offer "rotate" do this correctly; most desktop editors do not, and a rotate in an ordinary editor costs you a full generation.

Progressive JPEG is worth knowing: the same data reordered so a coarse version arrives first and refines. It is typically 2 to 10 per cent smaller than baseline and shows something useful earlier on a slow connection. There is no quality difference. Turn it on for anything going on the web.

MozJPEG, a free encoder from Mozilla, produces files roughly 5 to 15 per cent smaller than the standard library at matched visual quality by choosing better tables and trellis-optimising the coefficients. Squoosh in a browser uses it; `cjpeg` from the mozjpeg package uses it on the command line; it is free either way.

The honest limits

Quality numbers are not comparable between encoders. Quality 80 in libjpeg, in MozJPEG, in Photoshop's *Save As* and in Photoshop's *Save for Web* are four different things producing four different file sizes. Photoshop's *Save As* runs 0–12 and its *Save for Web* runs 0–100 and they do not correspond. Never specify a quality number in a brief without naming the tool; specify a file size or a visual test instead.

Artefacts cannot be removed. There are deblocking filters and AI restoration models, and what they do is smooth or regenerate, not recover. The coefficients were rounded to zero; the information is gone. The only fix for a badly compressed file is a better source file.

And the whole thing is aimed at photographs of the real world. JPEG's assumptions — smooth gradients, soft edges, colour that changes slowly — are true of a photograph and false of a screenshot, a chart, a line drawing or a logo. Using it on those is not a bad setting; it is the wrong tool.

BEFORE YOU MOVE ON

A designer saves a poster with thin red headline type on white as a JPEG. The type has a soft, dirty fringe. They raise the quality slider from 70 to 95 and the fringe is still there. Why?

A. The fringe is ringing from the cosine transform, and ringing is unaffected by the quantisation table.
B. The colour channels were stored at half resolution before any quantisation happened, so the quality setting never touches that loss.
C. Red is the channel JPEG compresses most aggressively, so red content always needs a different format regardless of settings.
D. The file was re-saved several times, and accumulated generation loss cannot be undone by a later high-quality save.

8 · 9 min

Some colours have no number

THE ONE THING TO KEEP

A colour space is a set of primaries defining a triangle of reachable colours, so converting to a smaller one either clips saturated areas into flat patches or compresses everything slightly, and the rendering intent is where you choose which.

What a colour space actually specifies

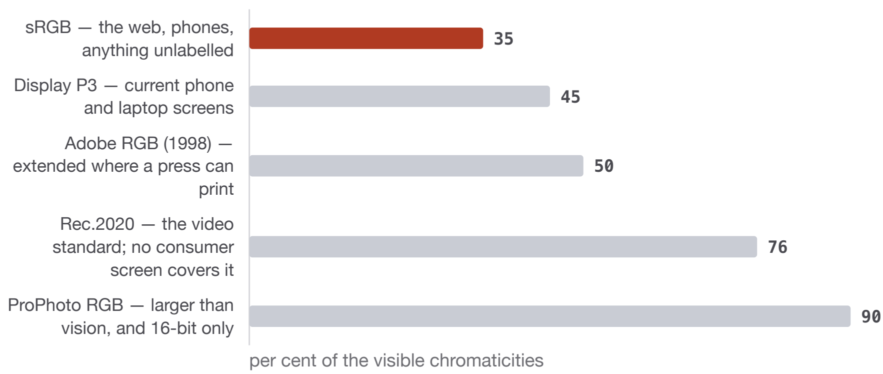
Saying a file is "sRGB" is saying three things at once:

1. **Primaries** — precisely which physical red, green and blue the values 255,0,0 / 0,255,0 / 0,0,255 refer to.
2. **White point** — what all three at maximum means. Almost everything uses D65, a standardised daylight around 6500 K.
3. **Transfer function** — the gamma curve from the earlier lesson.

Change the primaries and the same numbers mean different colours. That is the whole mechanism behind every "it looked different on their screen" conversation.

The space's **gamut** is the triangle those three primaries enclose on the chromaticity diagram — every colour you can make by mixing them. Anything outside the triangle has no representation in that space at all. Not a rounded one: none.

How much of what a person can see each space encloses



Areas on a chromaticity diagram, which is a statement about which colours, never about how bright. A wide space spreads the same 256 codes over more colour, so every step is a bigger jump: ProPhoto in 8 bits bands on ordinary skin, and its primaries lie outside human vision entirely.

The spaces you will meet, and how much bigger they are

- **sRGB** (1996) — the default for the web, for phones, for anything unlabelled. It covers roughly a third of the chromaticities a person can see. Its weak corner is saturated cyans, greens and oranges.
- **Display P3** — the space of modern phone and laptop screens, about 25 per cent larger than sRGB, mostly in reds and greens. Apple devices have used it since 2015 and current Android flagships do too.
- **Adobe RGB (1998)** — similar size to P3 but extended in a different direction, notably cyan-greens. It exists because those are the colours a CMYK press can print and sRGB cannot, which is why print workflows still use it.
- **ProPhoto RGB** — enormous, and deliberately so: its primaries lie *outside* the range of human vision, so some representable values correspond

to no real colour. Lightroom works in a variant of it internally.

- **Rec.2020** — the video standard for wide-gamut HDR. No consumer display covers it fully.

A concrete example: a saturated traffic-cone orange, or the cyan of a swimming pool, or a pure LED green sign. Photograph any of those and the sensor recorded them fine. Convert to sRGB and they hit the wall of the triangle and flatten into a single muddy tone with no internal detail — a shape that was a shaded cone becomes a flat orange blob.

What happens at the wall: rendering intents

When you convert from a larger space to a smaller one, the software has to decide what to do with the colours that do not fit. That decision is the **rendering intent**, and you choose it in the Convert to Profile dialogue.

- **Relative colorimetric** — keeps every in-gamut colour exactly where it is and clips the out-of-gamut ones to the nearest point on the boundary. Accurate for most of the picture, and it destroys detail in the saturated areas, because several distinct colours all clip to the same edge point. Turn on **black point compensation** with it, or your shadows will block up.
- **Perceptual** — compresses the whole gamut inwards so the relationships between colours survive. Every colour shifts slightly, including ones that would have fit. Better for an image with large saturated areas; unnecessary for one without.
- **Saturation** — keeps things vivid at the expense of accuracy. For charts and business graphics, not photographs.
- **Absolute colorimetric** — additionally reproduces the source white point, so paper white shifts. For proofing one press on another, not for delivery.

The practical rule: relative colorimetric with black point compensation by default; switch to perceptual when a soft proof shows you a large area going flat.

Soft proofing: seeing the loss before it happens

You can preview the conversion. In Photoshop, *View* > *Proof Setup* and *View* > *Gamut Warning* paints out-of-gamut pixels grey. GIMP has **View > Colour Management > Soft-proofing** with an out-of-gamut warning colour you can set. Both let you load an ICC profile for the destination — a printer's paper profile, or plain sRGB.

Do this before you promise a client a colour. Ten seconds of gamut warning prevents the conversation where a printed brochure's brand green is visibly duller than the website's.

ProPhoto and the 8-bit trap

A wide space spreads the same 256 codes over a much larger range of colour, so each step is a bigger jump. ProPhoto RGB in 8 bits is a known way to produce banding in ordinary skin tones, because the steps between adjacent codes are large enough to see.

The rule follows directly: **wide gamut requires more bits.** If you work in ProPhoto or Adobe RGB, work in 16-bit. sRGB in 8-bit is fine because the space is small enough that the steps stay under the visible threshold for most content.

Free tools, and the honest limits

GIMP has full ICC colour management, soft proofing and profile conversion. darktable and RawTherapee both let you pick a working profile and an output profile. **DisplayCAL** with **ArgyllCMS** drives calibration hardware and builds profiles, free. **littleCMS** is the engine underneath most of it.

The gap worth knowing: **Photopea's colour management is thin.** It largely assumes sRGB. For web work that is usually harmless; for print work, or for handling a client's Adobe RGB files, it will silently get things wrong. This is the clearest case in the course where the free browser tool is not a substitute and GIMP is.

And two limits on the whole idea. First, a wide-gamut file on a wide-gamut monitor still looks wrong if the application is not colour-managed — most browsers are now, most games and many desktop utilities are not, so the same file can look oversaturated in one window and correct in another on the same screen. Second, gamut is about *which* colours, not *how bright*. It says nothing about dynamic range, and a wide-gamut standard-range image is a completely different thing from an HDR one.

BEFORE YOU MOVE ON

A photograph of a bright cyan swimming pool, edited in Adobe RGB, is converted to sRGB with relative colorimetric intent. The pool comes out as a flat patch with no visible variation in the water. What happened?

A. Several distinct cyans all sat outside the sRGB triangle and were clipped to the same boundary colour, so the differences between them ceased to exist.

B. Relative colorimetric intent reduces saturation globally, and a perceptual conversion would have kept the original saturation intact.

C. The file should have been converted to 16-bit first, because 8-bit cannot hold enough cyan values to describe water.

D. Black point compensation was left off, which collapses the darker end of every channel including the water's mid-tones.

9 · 8 min

The monitor is a measuring instrument nobody calibrated

THE ONE THING TO KEEP

Calibration does not make your screen match anyone else's — it makes your own judgements repeatable and eliminates the display as a suspect — and the cheapest version of it is checking every delivery on three different devices.

Four ways the screen misleads you

Every editing decision you make is a comparison between what you see and what you want. If the seeing is unreliable, so is every decision, and the results arrive as complaints rather than as a visible fault on your own machine.

Brightness. A laptop at full brightness in a dim room is typically 300 to 500 cd/m². If you edit there, you will brighten the image less than it needs, because it already looks bright to you. The client opens it on a phone outdoors and it is dark. The standard for print work is around 120 cd/m²; for screen work 120 to 160 is sensible. On most laptops that is roughly half the brightness slider, and it will feel dim for a day and then feel normal.

White point and colour cast. Panels ship with wildly different white points, often bluer than the 6500 K standard because blue-white sells in shops. Edit on a blue-tinted screen and you will warm your images to compensate, and every one of them will look orange elsewhere.

Viewing angle. Cheap TN laptop panels shift tone with head position — tilt the lid five degrees and the shadows change. You cannot judge tone on one. IPS panels are stable and most laptops sold now have them; a two-minute test is to open a mid-grey image and move your head.

Ambient light. A window behind you puts reflections on the screen; a warm lamp beside it changes your eye's adaptation. Your visual system normalises to the room, not the screen. Working in a room with a strong colour on the walls will pull your colour judgement in the opposite direction all day.

What calibration hardware does

A colorimeter — a small puck that sits on the screen, around £110 to £150 for a SpyderX or i1Display — measures what the screen actually emits while software shows it known values. It then produces two things: adjustments to load into the graphics card, and an ICC profile describing the display's real behaviour so colour-managed applications can compensate.

DisplayCAL, driving **ArgyllCMS**, is free, open source, and drives almost every colorimeter on the market. The hardware costs money; nothing else has to.

The honest statement about calibrating by eye: you can correct an obvious colour cast with a visual test pattern, and you cannot measure luminance at all without an instrument. Your eye adapts within seconds, which is precisely the faculty that makes it useless as a meter. Visual calibration is better than nothing and is not calibration.

What to check without any hardware at all

Three free tests that catch the worst problems in five minutes.

1. **Shadow and highlight steps.** Display a greyscale step wedge — a strip of patches at 0, 2, 5, 10 ... 95, 98, 100 per cent. You should be able to distinguish 2 per cent from 0 and 98 from 100. If the dark steps merge, your black level or contrast is wrong and you will over-lift shadows in everything you edit.
2. **Neutral check.** Open a pure 128,128,128 grey filling the screen and look at it against a sheet of white printer paper under daylight. If the screen is obviously bluer or pinker than the paper, you know which direction your judgement is skewed.
3. **Uniformity.** The same mid-grey at full screen. Cheap panels are often visibly darker in the corners by 10 per cent or more. Judge tone in the middle of the screen, not at the edges.

Check the picture where the picture will live

The most useful habit in this lesson is also the cheapest: **look at the finished file on three devices before delivery.** Your monitor, a phone, and one other screen — a colleague's laptop, a TV, a tablet.

Phones deserve particular attention because they do things to your image after you deliver it. Auto-brightness changes exposure judgement continuously. True Tone and Night Shift warm the whole display towards the room. OLED panels push saturation. And a phone is where most of your audience will actually see the work.

If an image looks acceptable on all three, it will survive the fourth. If it looks right on one and wrong on the other two, the one is the outlier.

What calibration does not achieve

It does not make your screen match everyone else's. Nothing does. The viewer's phone is uncalibrated, at a brightness they chose, in a room you cannot see, possibly with a blue-light filter on.

Calibrating makes your own decisions **repeatable and defensible**: the same file looks the same to you next month, two machines in the same studio agree, and when a print comes back wrong the screen is eliminated as a suspect. That is the actual benefit, and it is worth having. Expecting more than that from it leads to the frustrated conclusion that calibration "does not work".

There is one place it does deliver a hard promise: with a calibrated display and a paper profile, a soft proof genuinely predicts a print closely enough to approve it without printing. For anyone doing print work, that alone pays for the puck in a month of wasted proofs.

BEFORE YOU MOVE ON

An editor works on a laptop at full brightness in a dark room. Clients repeatedly say the images arrive too dark. What is the mechanism, and what actually fixes it?

A. The laptop's colour profile is wrong, so installing the manufacturer's ICC profile will correct the delivered brightness.
B. The images are genuinely fine and the clients' screens are at fault, so the fix is to ask them to raise their brightness.
C. The editor sees a bright image on a bright screen and therefore lifts exposure less than needed, so the fix is to set the display near 120 cd/m² and re-judge.
D. Dark rooms cause the eye to lose shadow sensitivity, so working with the room lights on would correct the exposure judgement by itself.

10 · 9 min

Straight lines that are not straight, and colour that is not there

THE ONE THING TO KEEP

Geometry corrections resample the whole frame, so distortion, vignetting and lateral chromatic aberration go first, before cropping or retouching — and longitudinal fringing, flare and diffraction are losses no profile can undo.

The image arrives already distorted

Before any editing decision, the optics have altered the picture in four measurable ways. All four are correctable, all four are best corrected first, and knowing which is which stops you fighting the wrong one.

Geometric distortion. Straight lines bow. **Barrel** distortion bulges outward, common at wide angles; **pincushion** pinches inward, common at telephoto. A phone's ultrawide camera can have eight to ten per cent barrel distortion — enough that a doorframe at the edge of frame is visibly curved. Some phones already correct it in software before you see the JPEG; shoot DNG and you get the raw, bent version.

Vignetting. The corners are darker than the centre, because less light reaches the edge of the frame. Typically half a stop to two stops wide open, less when stopped down. Mild vignetting is often desirable and frequently added back deliberately; heavy vignetting on a product shot against white shows as grey corners on what should be a uniform background, which is a real nuisance.

Chromatic aberration. The lens focuses different wavelengths differently. Two kinds, and the difference matters:

- **Lateral (transverse) CA** — the channels are magnified by slightly different amounts, so you get coloured fringes that grow towards the corners, usually cyan-red or blue-yellow, along high-contrast edges. This is fully correctable, because the fix is simply to rescale the red and blue channels by a fraction of a per cent.
- **Longitudinal CA** — the channels focus at different distances, producing purple or green fringing on out-of-focus highlights, especially wide open. This is **not** geometrically correctable, because the channels are not displaced; they are differently blurred. The only treatments are desaturating the fringe colour locally or accepting it.

Diffraction. Stop a lens down past roughly f/11 on a full-frame camera, or f/8 on a crop sensor, and light bending round the aperture blades softens the whole frame. On a phone, with its tiny fixed aperture, this is already in play. Sharpening will not fix it, because it is a loss of resolution rather than a loss of contrast.

Profiles do the work for you

Every correction above is specific to a lens, and often to a focal length and aperture. Lens profiles are measured tables of those corrections.

Lensfun is the free, community-maintained profile database, and it is used by darktable, RawTherapee, digiKam and a GIMP plugin. It covers most common lenses and most phones. Adobe ships its own profiles in Camera Raw and Lightroom. In darktable the module is *lens correction*; in RawTherapee it is under *Transform > Lens/Geometry*.

When there is no profile for your lens — which happens often with phone cameras and new releases — every tool offers manual sliders. For distortion, find a straight line near the edge of the frame, turn on the grid overlay, and drag until it is straight. For lateral CA, most developers have an automatic option that works well without any profile, because the correction can be derived from the image itself by aligning the channels along edges.

Order matters, and the reason is arithmetic

Geometry corrections **resample every pixel in the frame**. They are the one class of edit that moves the coordinates of everything.

So do them first, before cropping, before retouching, before any local work. Three consequences follow:

- Retouching done before a distortion correction gets stretched with everything else. A cloned patch that matched perfectly can end up smeared.
- Distortion correction changes the frame shape, so the corners need trimming. You lose a few per cent of the image, and if you had already cropped tight, that loss comes out of your composition.
- Because correction stretches the corners more than the centre, **corner sharpness gets worse**. You are interpolating existing pixels over a larger area. This is real and unavoidable, and it is why heavy distortion correction on an already-soft ultrawide corner produces mush.

Perspective is not distortion

A commonly confused pair. Point a camera up at a building and the verticals converge. That is not a lens fault; it is correct projective geometry, exactly what your eye does. The building really does subtend a narrower angle at the top.

Correcting it — **keystone correction**, *Upright* in several tools, *Perspective* in GIMP's Transform tools, *Transform* > *Perspective Warp* in Photoshop — is a deliberate choice to draw the building the way architectural drawings do. It works by stretching the top of the frame, which costs resolution at the top and crops the sides. Correct a 15-degree upward tilt and you can lose a fifth of the image. For serious architectural work the answer is a tilt-shift lens or shooting wide and straight and cropping, not a heavy warp.

Hugin, free and open source, does this and far more — it is a panorama tool whose lens and perspective correction engine handles cases no slider will.

What cannot be corrected

- **Flare and veiling haze** from a light source in or near the frame. Contrast recovery helps; the scattered light genuinely landed on the sensor and is now part of the signal.
- **Longitudinal CA**, as above.
- **Diffraction softness** — you can add acutance with sharpening, and the resolution is gone.
- **Extreme fisheye stretch at the very corner**, where the sampling density was so low that correction has almost nothing to work with.

And one more honest point: correcting distortion is not always right. A 24 mm portrait has enlarged the nose, and "fixing" the geometry does not fix that, because it is perspective from a close camera position rather than lens distortion. The answer there is to stand further back and use a longer lens. Optics problems are usually cheaper to solve at capture than in the editor, which is a theme you will meet again in this course.

BEFORE YOU MOVE ON

A photograph taken wide open shows purple fringing on the out-of-focus highlights behind the subject. Enabling the lens profile's chromatic aberration correction removes the coloured edges at the frame corners but leaves the purple highlights untouched. Why?

A. The lens profile covers only the corners, and a manual correction applied to the centre of the frame would remove the purple.
B. The correction works by rescaling channels that are displaced sideways, and the purple comes from channels that are differently focused rather than differently magnified.
C. Purple fringing is sensor blooming rather than a lens effect, so no optical correction of any kind applies to it.
D. The profile was built at a different aperture, so re-running the correction with the correct aperture recorded in EXIF would remove it.

CASE STUDY · MODULE 1

The bank's logo, the banner, and the file nobody kept

A two-person print shop in Erode, three days before a co-operative bank's sixtieth anniversary function.

The order

The branch manager wanted a vinyl banner three metres wide for the front of the building, with the bank's crest, its name in English, and its name in Tamil underneath. He emailed the logo. It was a PNG, 900 pixels across, saved from the bank's own website some years earlier by somebody who no longer worked there.

The shop's owner did the arithmetic on the back of the order slip. Three metres is about 118 inches. Nine hundred pixels across 118 inches is under eight pixels per inch.

That number by itself does not condemn the job. Billboards are routinely output at ten to twenty pixels per inch and nobody standing on the road notices, because a photograph read from twenty metres has no detail at that scale anyway and degrades gently. The rule of thumb that says 300 is for something held in the hand.

But this was not a photograph. It was a crest with a fine rule around it, a serif wordmark, and Tamil letterforms whose vowel marks are small strokes sitting above and beside the consonants. Hard edges and small strokes are exactly what a low sample count destroys, and a guest standing two metres from the entrance would be reading it at arm's length, not from a moving car.

Three routes, none of them free

Upscale it. Upscayl, free, twenty minutes, four times or eight times. On skin and brickwork it would have been remarkable. On letterforms it predicts plausible shapes from the low-resolution evidence, and plausible is not the same as correct. The result looks crisp, which is the dangerous part, because nothing in the file announces that the strokes have been invented.

Redraw it. The crest could be traced as vector in Inkscape in about four hours, and vector has no resolution at all, so three metres or thirty would be the same file. The Tamil wordmark was the problem: the original was set in a licensed font nobody in the shop had, so it would have to be traced letter by letter by eye. That is a full day, and a full day is the wedding-card job that was already on the bench.

Ask head office. One phone call and, if somebody answered, the original artwork. The bank's marketing department is in Coimbatore and had not replied to an email about a rate card in two weeks. The function was on Friday and vinyl printing needed the file by Thursday morning.

The cost sat on every side. A day of a two-person shop's capacity is a real loss. So is a banner with a slightly wrong crest, at a function the district collector was attending, produced by a shop whose name would be on the bottom edge.

What the owner actually decided

He ran all three at once, which is the decision worth noticing. He put his assistant on the Inkscape trace on Tuesday morning as insurance, sent the upscaled version to the large-format printer as a single A3 proof so he could look at it rather than argue about it, and telephoned Coimbatore three times until somebody found the artwork.

The A3 proof settled the argument on its own. At one metre the English wordmark looked acceptable. The Tamil vowel marks were visibly the wrong shape — one had acquired a curl it does not have — and the fine rule around the crest had gone slightly wavy where the model had smoothed it.

> **STOP HERE**
>
> Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

Coimbatore sent an EPS on Wednesday afternoon, four hours after the assistant had finished the crest and started on the wordmark. The banner printed from the vector artwork and nobody at the function thought about it, which is the point. The shop kept the half-finished trace, and it has been used twice since for signage where head office was slower.

Two things changed permanently. The order slip now has a line on it that says *vector artwork received: yes / no*, filled in when the job is taken rather than when it is due, so the conversation happens on day one instead of the night before. And the shop keeps a folder of client logos in whatever the best available form is, backed up to a pen drive kept at the owner's house, because the reason this job nearly failed is that the only copy in existence at the branch was a screenshot of a website.

Billboards are printed at ten to twenty pixels per inch and nobody complains. Why was eight pixels per inch a problem on this banner?

Because viewing distance and content both matter, and this job had the wrong values for both. The billboard rule holds for photographs read from far away: a photograph has no hard edges, so missing detail reads as softness and softness is invisible at distance. A crest with a fine rule and a Tamil wordmark is made almost entirely of hard edges and small strokes, and it would be read by guests standing a couple of metres from the door. The ppi figure in a file is only a tag and changes nothing; what matters is pixels divided by the printed size, checked against how close the reader will stand.

The upscaled version was not merely soft. What made it dangerous, and how did the proof expose that?

A model-based upscaler does not interpolate between the samples you have; it predicts a plausible high-resolution image from the low-resolution evidence. On photographic texture the plausible answer is usually close to the true one. On letterforms it is not: the model regenerates the strokes, and they come back crisp and subtly the wrong shapes, with no warning that anything was invented. Printing one A3 proof and looking at it at reading distance turned an argument about a setting into an observation — a Tamil vowel mark with a curl it does not have, and a wavy rule where the model had smoothed a straight line.

The shop added one line to its order slip afterwards. Why is that worth more than any technique in this lesson?

Because it moves the limit to the start of the job, where it can still be acted on. Opening the file, noting the pixel dimensions and dividing by the printed size takes under a minute, and it tells you on the day the order is taken whether the job is possible. Saying so on day one is a professional conversation; discovering it on the night before the deadline is an emergency in which every remaining option costs somebody something. The second change — keeping client artwork in the best form available — attacks the actual cause, which was that the only copy anyone could reach was a screenshot of a web page.

MODULE 1 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A client sends a 600-pixel-wide product photograph for a listing that needs 2,400 pixels. Enlarging it in any editor gives a soft result. What is the mechanism?

A. Bicubic is the wrong filter here, and Lanczos would recover the missing detail
B. The file is tagged 72 ppi and needs to be re-tagged at 300 before it is resized
C. Interpolation can only average samples that exist
D. The JPEG compression has to be decoded and removed before a resize can work

2 You want to build a mask from a single colour channel and need the one carrying the least noise. Why is green usually it?

A. Sensors carry twice as many green photosites as red or blue
B. Green sits in the middle of the histogram, so it clips last
C. JPEG stores green at full resolution and the other two at half
D. Green is weighted at 0.7152 for luminance, which suppresses its noise

3 You open an 8-bit JPEG of a smooth sky, convert the document to 16 bits per channel, and then apply a strong Curves stretch. What has the conversion achieved?

A. Recovered levels the JPEG encoder discarded
B. Stopped later edits rounding onto the coarse grid
C. Nothing at all, because the stretch bands exactly as it would have in 8-bit
D. Removed the need to dither, because 65,536 levels never band

4 A photograph of a starfield, downscaled to a quarter of its width, comes back noticeably darker than the original. What has gone wrong?

A. The resize filter clipped the highlights on the bright points
B. Lanczos ringing has darkened the pixels around each bright point
C. The image was converted from Adobe RGB to sRGB without a profile
D. Stored values were averaged instead of the light they represent

5 Red type on a white background, saved as JPEG, has a soft dirty fringe. Raising the quality from 70 to 95 barely changes it. Why not?

A. The fringe comes from chroma subsampling, which happens before quantisation
B. Quality 95 is still lossy, so some fringing survives at every possible setting
C. The ringing is a property of the 8 by 8 transform and cannot be reduced at all
D. The red channel clipped at capture and the encoder is preserving that clipping

6 A raw developer recovers about a stop of a slightly blown sky that the camera's own warning said was gone. Which fact makes that possible?

A. A raw file stores a second, darker exposure alongside the main one for recovery
B. Raw files are 14-bit, so there is always headroom above white whatever the exposure
C. Two channels still hold real values when one has clipped
D. Demosaicing can be re-run with an algorithm that refuses to clip any channel

MODULE 2

The non-destructive workshop

A layered file is a recipe, not a picture. This block builds the workshop you will do everything else in: what a blend mode computes, why the order of the stack changes the answer, what a group actually does to rendering, smart objects and linked files, the Channels panel as storage and arithmetic, versions and backups that survive a client changing their mind twice, and what to do when the file gets too heavy for the machine you own.

> BY THE END OF THIS MODULE YOU CAN
>
> Build and hand over a layered document another editor can take apart — correctly ordered, grouped, masked, named and versioned — and diagnose from the stack alone why an adjustment is affecting the wrong layers or why the file has become too slow to work in

---

11 · 9 min

Every edit should still be undoable next week

THE ONE THING TO KEEP

Stack instructions on top of the pixels instead of changing them, and save the layered file before you save anything else.

The mistake that ends the job

You open the photo, brighten it, warm it, crop it, flatten it, save the JPEG over the original and send it. The client says: lovely, slightly darker, and can we see it uncropped.

Now you are stuck. The cropped-away pixels are gone. The tonal range you stretched is gone. Every further edit is applied to the damage.

Non-destructive editing is the discipline of never modifying original pixels — only stacking instructions on top of them. It is not a preference for tidy people. It is what makes a second revision cost thirty seconds instead of an afternoon.

The four mechanisms

Adjustment layers. A Curves or Hue/Saturation layer sits above the photo and changes how it is displayed, not what it contains. Double-click it a week later and the sliders are exactly where you left them. In Photoshop: the half-filled circle at the bottom of the Layers panel. Photopea, Krita (it calls them *filter layers*) and Affinity Photo all have the same thing in the same place.

Layer masks. A greyscale channel attached to a layer that decides where it shows. That is the next lesson, and it is the important one.

Smart objects. Right-click a layer, *Convert to Smart Object*. Now the original is stored inside a container. Scale it down to 20 per cent and back up and it comes back sharp, because the resize is a display instruction applied to the stored original rather than a resampling of what is on screen. Filters applied to it become re-editable *smart filters*. Photopea supports smart objects. Affinity has an equivalent in its live filter layers.

Parametric editors. Lightroom, Capture One, and the free **darktable** and **RawTherapee** never write to your file at all. They record a list of edits in a sidecar and re-apply it every time you open the image. This is the most non-destructive model there is, and it is why serious work starts in a raw developer and moves to a pixel editor only for things a slider cannot do.

What each editor actually gives you

The money question. Photoshop is named in job adverts, so learn its vocabulary. But for this particular set of features:

- **Photopea** (photopea.com) runs in a browser tab, is free with advertisements, and opens and saves layered PSD. It has adjustment layers, layer masks, smart objects, Curves, blend modes. For non-destructive work it is genuinely equivalent to Photoshop for most jobs. Honest limits: everything lives in your browser's memory, so a 40-layer 100-megapixel file will crawl or crash; its colour management is thin, which matters for print; and it is maintained by essentially one developer, which is remarkable and also a risk.
- **GIMP** has excellent layer masks and channels and, since version 3.0 in early 2025, non-destructive layer effects. It still has no true adjustment layers, so a Curves change is usually baked into a duplicated layer. That is a real gap, and the workaround — duplicate, adjust, mask — costs you re-editability.
- **Krita** has filter layers and filter masks and is fully non-destructive. It is built for painting, so its photo-retouch tools are thinner. Excellent on a drawing tablet.
- **Affinity Photo** has adjustment layers, live filters and a fast engine. It was a one-off purchase of roughly £70 with no subscription; after Canva bought Serif the app was relaunched free of charge in late 2025 with some newer features tied to an account. That corner has moved twice already, so check the current terms rather than trusting this paragraph.
- **On a phone**: Snapseed keeps every edit as a re-editable *stack* you can re-open and change, which is real non-destructive editing on a device that costs nothing extra. Lightroom mobile's free tier is parametric by design.

More on picking between them in choosing your tools.

The file discipline that costs nothing

- Save the layered working file **first**, before you export anything. PSD in Photoshop and Photopea, XCF in GIMP, KRA in Krita, AFPHOTO in Affinity. The JPEG is a delivery format, never a working one.
- Name versions. `sandal-front_v3.psd`, not `final_final_2.psd`. You will need v2 back.

- Never save a JPEG over a JPEG. Every save re-compresses what the last one already damaged.
- Keep the camera original in a folder you do not edit in.

What this does not save you from

Non-destructive is not infinite. Three honest limits.

Bit depth still bites. An 8-bit file has 256 levels per channel. Stretching it hard leaves gaps between them, and gaps show up as banding in skies. Adjustment layers delay the damage until you flatten, but they do not create data. Work in 16-bit — **Image > Mode > 16 Bits/Channel** — for anything with a big tonal stretch.

The file gets heavy. A 30-layer smart-object stack can be a gigabyte and slow on an old laptop. Rasterise the parts you have genuinely finished.

Someone else has to open it. A PSD full of smart filters opens correctly in Photoshop and Photopea and only approximately in GIMP. If you are handing off, also export a flattened TIFF.

Today: open your last edited photo. If there is no layered file next to it, you did the work twice already and do not know it yet.

BEFORE YOU MOVE ON

A retoucher brightens a photo with Curves applied straight to the pixel layer, flattens, saves a quality-90 JPEG and sends it. The client asks for it slightly darker and less yellow. Editing that JPEG produces blotchy bands in the sky. What is the best explanation?

A. Quality 90 was too aggressive; exporting at quality 100 would have avoided the banding on the second edit.
B. The first edit baked a stretched tonal range into 8 bits, leaving gaps between the surviving levels, and the second edit stretches what is left until those gaps become visible steps.
C. The monitor is uncalibrated, so the bands are a display artefact rather than something in the file.
D. The file should have been saved as PNG, which is lossless, and the second edit would then have been clean.

12 · 10 min

The eraser is the tool that costs you the second revision

THE ONE THING TO KEEP

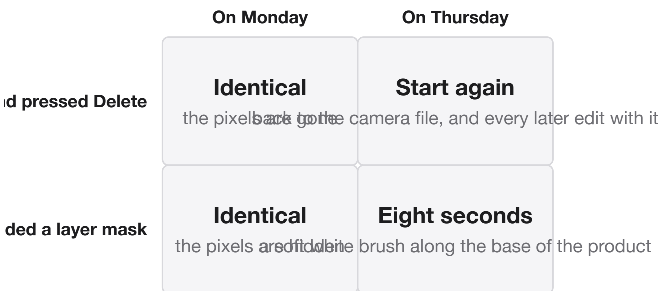
Black conceals, white reveals — mask instead of erasing, and every revision becomes a brushstroke rather than a restart.

Deleting and hiding look identical until Thursday

Two people cut a product out of its background. One selects the background and presses Delete. The other selects the product and clicks **Add Layer Mask**. On screen the results are the same.

On Thursday the client says the shadow under the product was nice, could we keep a bit of it. The first person starts again from the camera file. The second person picks up a soft white brush, paints for eight seconds, and sends it back.

Two cut-outs that look identical until the revision arrives



The client asks for a little of the shadow under the product back. Nothing about the two files looked different on Monday, and the whole difference between them is whether the pixels were deleted or hidden. Black conceals, white reveals, and the revision becomes a brushstroke.

This is the single skill that separates people who can edit from people who own a copy of Photoshop.

What a mask actually is

A layer mask is a greyscale image attached to a layer, the same pixel dimensions as the layer, with one value per pixel from 0 to 255.

- **Black hides.** 0 means that pixel of the layer is fully invisible.
- **White reveals.** 255 means fully visible.
- **Grey is partial.** 128 means half opacity, pixel by pixel.

Nothing is deleted. The pixels are all still sitting there, and the mask is just a stencil deciding what gets through. Paint the stencil white and they come back.

Say it once as a sentence and it will stay: *black conceals, white reveals*.

Two things to know immediately. Press \ (backslash) in Photoshop to see the mask as a red overlay on the image, or Alt/Option-click the mask thumbnail to see the mask on its own in black and white — that is how you find the ragged edge you cannot see against the picture. And press **X** to swap your brush between black and white, because masking is nothing but painting and unpainting.

The five things you do to masks

1. **Paint them.** A soft round brush at 30 to 50 per cent opacity, built up in passes. Hard brush for hard edges, soft for hair and shadow.
2. **Gradient them.** Drag a black-to-white gradient across a mask on a Curves layer and you have darkened a sky exactly the way a photographer's graduated filter does, with no line where it stops.
3. **Feather them.** A mask with a one-pixel transition looks cut out with scissors. Two pixels of feather is usually the difference between fake and not.
4. **Invert them.** Ctrl/Cmd+I. You almost always select the easier thing and then invert.
5. **Disable them.** Shift-click the thumbnail to switch a mask off and check what you have hidden. Do this before you deliver.

Masks on adjustments are the real workflow

The moment that makes it click: every adjustment layer arrives with a mask already attached.

So darkening one face in a group photo is not a selection problem and not a dodge-and-burn problem. It is: add a Curves layer, pull the curve down, press Ctrl/Cmd+I to fill the mask with black so the darkening is hidden everywhere, then paint white with a soft brush over that one face. Too strong? Lower the layer opacity. Wrong face? Paint black.

Nothing was changed in the photograph. Every part of that is reversible for as long as the file exists.

Two more masks with confusing names

Clipping masks

A different thing with a confusingly similar name. A clipping mask makes a layer show only where the layer directly beneath it has pixels. Alt/Option-click the line between two layers in the Layers panel, or Ctrl/Cmd+Alt+G.

Use it when you want a texture, a colour fill or an adjustment to affect exactly one layer and nothing under it — for example, a Hue/Saturation layer clipped to a cut-out shirt so you can recolour the shirt and only the shirt. Photopea, Krita and Affinity all support this. GIMP does it through *Layer* > *Mask* > *Add Layer Mask* plus grouping, which is clumsier.

Luminosity masks: the selection with no edge

A hard-edged selection always risks a visible seam. Sometimes you want to affect *the bright parts* rather than a region.

In Photoshop or Photopea, open the Channels panel and Ctrl/Cmd-click the RGB thumbnail. You get a selection whose strength at every pixel equals that pixel's brightness. Now add a Curves layer: the adjustment automatically applies fully to highlights, partly to midtones, not at all to shadows, with no boundary anywhere. Invert it to target the shadows instead.

This is how you recover a blown sky without a halo around the roofline, and it costs two clicks. GIMP does the same by adding a layer mask with the *Grayscale copy of layer* option — one of the places where the free tool is not worse at all.

The honest limits

A mask cannot invent edge detail that the photograph does not contain. If the subject is motion-blurred against the background, no mask separates them cleanly, because there is no boundary in the pixels to find — the edge pixels are genuinely a mixture of both.

Masks also cost memory. A dozen high-resolution masks in an 8-bit file is fine; the same in 16-bit on a four-gigabyte laptop is not. And a mask painted at 200 per cent zoom often looks crude at 100 per cent. Check at fit-to-screen before you export, then at 100 per cent along the edge.

Today: take any photo, add a Curves layer, drag the middle of the line down until it is clearly too dark, invert the mask, and paint the darkening back in only where you want it. That is the whole craft in one gesture.

BEFORE YOU MOVE ON

Someone cuts a product out by selecting the background and pressing Delete on the background layer, then flattens and saves. The client now wants a little of the original contact shadow at the base kept. What is the situation?

A. Those pixels no longer exist in the file, so the only route back is re-cutting from the camera original; a layer mask would have made it a few seconds of white brush.

B. Adding a layer mask to the flattened file now and painting white will bring the shadow back, since masks control visibility.

C. The eraser opacity was set too high; lowering it and repeating the cut-out is the fix.

D. The undo history holds the state before the delete, so stepping back through it recovers the shadow.

13 · 10 min

Twenty-seven blend modes, six ideas

THE ONE THING TO KEEP

Every blend mode is a short per-channel formula, so white vanishes in Multiply, black vanishes in Screen, mid-grey vanishes in Soft Light, and setting a tonal adjustment to Luminosity separates contrast from the saturation shift that normally comes with it.

Each mode is a formula run per pixel, per channel

The blend mode dropdown is intimidating because it is a long list of names with no obvious logic. Underneath, each one is a short piece of arithmetic taking the pixel below (the **base**) and the pixel above (the **blend**) and producing a result. Learn six formulas and the list collapses.

Work in 0-to-1 rather than 0-to-255 and the maths reads more cleanly.

Multiply: `result = base × blend`. Anything times 1 (white) is unchanged; anything times 0 (black) is black. So **white is invisible in Multiply** and everything else darkens. This is the mode for shadows, for a scanned signature on paper, for ink on a mockup.

Screen: `result = 1 − (1 − base) × (1 − blend)`. Multiply performed on the inverted values, then inverted back. **Black is invisible in Screen** and everything else lightens. This is the mode for light: flare, glow, fireworks on a night sky, smoke on black.

Overlay and Soft Light: conditional. Where the base is dark they multiply, where it is light they screen. The result is that contrast increases around a pivot. In both, **mid-grey (128) is the neutral value** and leaves the base unchanged. That fact is the entire basis of the dodge-and-burn layer technique you will meet in the tone module: fill a layer with 50 per cent grey, set it to Soft Light, and painting slightly lighter or darker grey lightens or darkens the picture underneath with nothing destroyed.

Difference: `result = |base − blend|`. Identical pixels give black. Useless as an effect and invaluable as a tool — stack two versions of a photograph, set the top to Difference, and anything that is not pure black is a place where they differ. This is how you align two frames precisely, and how you check whether an "identical" re-export really is identical.

Colour and Luminosity: these two split the image rather than combining it numerically. Luminosity takes the brightness of the blend layer and the colour of the base; Colour does the reverse. Set a Curves layer to **Luminosity** and it changes tone without the saturation shift that a tonal move normally causes. Set a retouching layer to **Colour** and you can repair a colour blotch without touching the skin texture underneath.

Hue and Saturation modes complete that family and are used far less often.

Why the contrast modes shift colour, and the fix

Apply a strong S-curve and the colours get more saturated as well as more contrasty. That is not a feature somebody added; it falls out of the arithmetic.

Steepening the curve moves each channel independently, and moving three channels apart from each other by different amounts is, by definition, an increase in saturation.

If you want the contrast without the colour shift, set the Curves layer's blend mode to **Luminosity**. The tone change goes through and the hue and saturation of the base are kept. This one habit fixes the commonest complaint about heavy editing — that the skin went orange — without a single extra adjustment.

Opacity and Fill are not the same control

Two sliders sit next to each other and most people never learn the difference.

Opacity fades everything about the layer, including any layer styles attached to it. **Fill** fades the layer's pixels but leaves layer styles at full strength. Drop Fill to 0 on a shape with a drop shadow and you get a floating shadow with no object. It is a specialist control, and knowing it exists saves an argument with a file that is behaving strangely.

For most blend-mode work, Opacity is the dose control: pick the mode for the effect, then dial the strength.

The free tools all have this

GIMP, Krita, Photopea and Affinity all implement the standard set, and most of them match Photoshop's results closely because the formulas are published. Krita has the largest list of all, including several that exist nowhere else.

Two real differences to watch for. GIMP since 2.10 offers both "legacy" and current versions of several modes, because it moved to computing them in linear light — the same issue from the gamma lesson — and the two give visibly different results for Overlay and Soft Light in particular. And **Soft Light is not one formula**. Photoshop, the W3C compositing specification and older GIMP each use a slightly different curve. A file blended in one application and opened in another can differ, subtly, in exactly those layers. If a layered file has to survive a handover, flatten the Soft Light layers or note what made them.

A short catalogue you will actually use

- **Multiply** — shadows, ink, darkening a texture onto a surface.
- **Screen** — light, flare, glow, dust and rain plates shot on black.
- **Soft Light** — dodge and burn on a grey layer; gentle contrast; High Pass sharpening.
- **Overlay** — the harder version of the same, and the standard for High Pass sharpening.
- **Colour** — recolouring an object, fixing a colour cast on one region.
- **Luminosity** — tonal adjustments without saturation drift.
- **Difference** — alignment and comparison, then switch it back off.

The honest limit

All of these run on the stored, gamma-encoded numbers, per channel, with no knowledge of what the picture contains. Multiply is a reasonable model of a filter absorbing light and Screen is a reasonable model of two lights adding, but only in linear space — run on encoded values, as they are by default, they are approximations that look plausible rather than physical simulations.

The practical consequence is that a composite built entirely out of blend modes will rarely look truly photographic on its own. Blend modes get you to a result quickly; matching light, shadow, grain and edge quality is what makes it convincing, and that is the compositing module later in this course.

BEFORE YOU MOVE ON

A retoucher applies a strong S-curve to a portrait and the skin becomes noticeably more orange as well as more contrasty. Setting the Curves layer's blend mode to Luminosity removes the colour shift. What does that reveal about the cause?

A. The curve was applied to the red channel only, and Luminosity mode redistributes the change across all three.
B. Luminosity mode reduces the strength of the adjustment, so the colour change is still present but below the visible threshold.
C. The monitor's profile exaggerates saturated skin tones, and Luminosity mode bypasses colour management.
D. Steepening the curve moved the three channels apart by different amounts, which is itself an increase in saturation, and Luminosity keeps the base image's colour while taking only the new brightness.

14 · 9 min

The same three edits in a different order give a different picture

THE ONE THING TO KEEP

Image operations are not commutative, so a layer stack is an ordered recipe — repairs under tone, tone under colour, sharpening last — and a stamped flatten is a snapshot that silently goes stale the moment anything beneath it changes.

An adjustment layer sees everything under it

The rule is short: an adjustment layer applies to the composite of every layer beneath it in the same group, and to nothing above it. Move it up the stack and it starts affecting more; move it down and it affects less. Clip it (from the masks lesson) and it affects exactly one.

That means the stack is not a pile of independent settings. It is an ordered recipe, and the order is part of the result.

Prove it with two numbers

Take a pixel at RGB 200, 100, 100 — a mid-toned warm red.

Path A: saturation first, then a darkening curve. Raise saturation, and the channels spread apart, say to 220, 80, 80. Now apply a curve that scales everything to 70 per cent: 154, 56, 56.

Path B: the darkening curve first, then saturation. Scale to 70 per cent: 140, 70, 70. Now raise saturation by the same amount, and the spread is applied to a smaller range: roughly 154, 63, 63.

Different answers, same two operations. This is not a rounding artefact; most image operations are not commutative, because each one is a non-linear function of the last one's output. When somebody says "I did exactly what you did and it looks different", the order of the stack is the first thing to check.

A stack order that works for nearly every job

Bottom to top. Each level exists because what is above it depends on what is below being settled.

A stack order that works for nearly every job

Output — the flattened copy that gets resized, sharpened, converted	Switched on last, or made fresh at each export. Sharpening exaggerates whatever it is handed, so nothing may come after it.
Effects — vignette, grain, glow	Applied over everything below so one treatment covers the whole picture rather than one element of it.
Colour — white balance correction, Hue/Saturation, the grade	Above tone, because a grade built on an unsettled exposure will not transfer to the next frame in the set.
Tone — global Curves, then masked local Curves and dodge and burn	Set the black and white points, then shape. Set a Curves layer to Luminosity and the contrast goes through without the saturation drift.
Repair — clone and heal on an empty layer, Sample All Layers on	Below tone, so a later curve change does not strand repairs that were sampled against different tones. Put it above the background and it adapts.
Background and geometry — the developed file, lens and perspective already applied	Geometry resamples every pixel in the frame, so it goes before any local work or it stretches your repairs with everything else.

An adjustment layer sees the composite of everything beneath it and nothing above, so the stack is an ordered recipe rather than a pile of settings. Two of these placements are load-bearing: repairs under tone, and sharpening above everything and applied last.

1. **Background** — the developed image, or the smart object containing the raw file.
2. **Geometry** — perspective, straightening, lens correction. Ideally already done in the raw developer, because these resample everything.
3. **Repair** — clone and heal layers, with *Sample All Layers* on. Damage should be gone before you judge tone, or you will be setting exposure around a dust spot.
4. **Global tone** — Levels or Curves for black point, white point and overall contrast.
5. **Local tone** — dodge and burn, masked Curves layers for individual regions.
6. **Colour** — white balance corrections, Hue/Saturation, selective colour, grading.
7. **Effects** — vignette, grain, glow.
8. **Output** — the flattened copy that gets resized, sharpened and converted. Kept in the master file as a group you switch on last, or made fresh each

time you export.

The two placements that are genuinely load-bearing:

Repair below tone. If you clone at the top of the stack, your repairs are made against the already-adjusted image. Change the curve afterwards and the repairs no longer match, because they were sampled from different tones. Put the clone layer directly above the background, under everything tonal, and it adapts.

Sharpening above everything, and applied last of all. Sharpening exaggerates whatever it is given. Applied under a contrast curve, the curve amplifies the halos it created. This is the same argument as the output lesson later, one level up.

Name the layers, or the file is write-only

A 30-layer file where every layer is called "Layer 14 copy" is a file only you can use, and only this week. Double-click the name and type four words: `burn – left wall`, `heal – dust on lid`, `curve – overall contrast`.

Colour-code groups if the tool supports it. Photoshop, Photopea and Krita all allow a colour label on a layer; use one colour for repairs, one for tone, one for output.

The test: could a colleague open this at 9 a.m. tomorrow, switch off your colour work, and see the untouched repair underneath? If not, you have a picture rather than a working file.

Stamp visible, and what it costs

Ctrl/Cmd+Alt+Shift+E creates a new layer containing a flattened copy of everything currently visible, leaving the stack beneath it intact. It is the standard move before a filter that cannot work on a live stack — most sharpening, frequency separation, and anything in GIMP where a filter needs real pixels.

The cost is honest and worth stating. A stamped layer is a snapshot. Change anything below it and the stamp does not update; you now have two versions of the picture in one document, and the visible one is the stale one. If you stamp, do it near the end, name it clearly, and expect to delete and re-stamp when you revise. Several long arguments with a client end with the discovery that the exported file came from a stamp made before the last round of changes.

What the free tools do differently

GIMP has no true adjustment layers, which makes this whole model harder. The standard workaround is to duplicate the layer, apply the adjustment destructively to the copy, and mask it — which gives you a maskable, removable, re-orderable layer with one loss: you cannot reopen its settings. GIMP 3.0 added non-destructive layer effects, which closes part of the gap. Plan the stack the same way and accept that revising a tone move means redoing it.

Krita has filter layers and filter masks and behaves much like Photoshop for this.

Photopea has adjustment layers, groups, clipping and stamp-visible with the same shortcuts, and reads and writes PSD, which makes it the closest free match for collaborative work.

One broader limitation: a deep stack is a record of your reasoning and also a performance cost, because every change re-composites everything above it. Twenty adjustment layers on a 50-megapixel file will lag on any machine. The next lessons in this block are about that trade.

BEFORE YOU MOVE ON

An editor clones out several dust spots on a layer placed at the very top of the stack, above all the tonal adjustments. They then change the main Curves layer to brighten the image, and the cloned patches now show as visible dark discs. Why?

A. The clone layer was sampled from the adjusted composite, so its pixels are fixed at the old brightness while everything beneath it moved.
B. Cloning always produces a slight tonal mismatch, and the Healing Brush would have avoided the problem wherever the layer sat in the stack.
C. The clone layer's blend mode reverted to Normal, so it is no longer inheriting the tone of the layers below it.
D. The Curves layer is below the clone layer, so it cannot reach the cloned pixels, and moving it to the top of the stack would fix the mismatch.

15 · 8 min

Why an adjustment stopped working when you put it in a folder

THE ONE THING TO KEEP

A group defaults to Pass Through, where adjustments inside still reach the layers below, and giving the group any other blend mode or an opacity below 100 isolates it into its own buffer so those adjustments become local.

The symptom that sends people to forums

You have a working file. A Curves layer near the top darkens the whole image. You tidy up by dragging a few layers into a group, and the darkening changes strength — or stops reaching the layers underneath entirely. Nothing was edited. The behaviour changed because a group is not only a folder.

Pass Through against Normal

A group has a blend mode like any layer, and its default is a mode no ordinary layer has: **Pass Through**.

Pass Through means the group is purely organisational. Everything inside renders as though the group were not there, so an adjustment layer inside the group still reaches down through the group's floor to affect the layers below it.

Normal — or any other blend mode — makes the group **isolated**. The contents are composited together into their own private buffer first, and only that finished result is blended onto the layers below. An adjustment layer inside now only affects the other layers in the group. It cannot see out.

That is the whole mechanism, and it explains the symptom exactly. Setting a group to Normal to give it a 60 per cent opacity — a perfectly reasonable thing to want — silently converts every adjustment inside it from global to local.

It cuts both ways, and the second direction is the useful one. If you want a colour grade to apply to just one part of a composite, put those layers and the grade in a group and set the group to **Normal**. The grade is now fenced in, with no clipping and no masking needed.

Group masks apply to everything inside

A mask on a group is applied to the group's composite result. Paint black on it and everything in the group is hidden there, regardless of each layer's own mask.

This is the cleanest way to build a complex local edit: three or four adjustment layers that together do something specific, in a group, with one mask on the group defining where it happens. Revising the region means painting one mask rather than four.

Be aware that layer masks inside the group and the group's own mask multiply. A pixel at 50 per cent on a layer mask inside a group whose mask is also 50 per cent at that point shows at 25 per cent. Chains of soft masks can fade away faster than you expect, and this is usually the reason.

Clipping inside groups

A clipping mask makes a layer show only where the layer directly beneath it has pixels. Inside a group, clipping works against the layer below within that group, and a clipped stack cannot reach outside it.

The useful combination: a cut-out product layer, with a Hue/Saturation layer clipped to it and a Curves layer clipped on top of that, all in one group. The two adjustments hit the product and only the product, the group holds them together, and the group's mask can then remove the whole arrangement from part of the frame.

The bottom layer of a clipping chain is the **base**, and its transparency defines the shape. If you later rasterise or fill that base layer edge to edge, everything clipped to it suddenly applies everywhere. That is the second forum question this lesson exists to answer.

Nesting, and where to stop

Groups nest. Three levels is usually the practical maximum before navigating costs more than the tidiness saves. A workable convention for a composite:

```
[group] OUTPUT          (sharpen, resize copy — off while work-
ing)
[group] GRADE           (global colour, vignette)
[group] SUBJECT         (cut-out + clipped adjustments + shad-
ow)
[group] BACKGROUND      (plate + its own tone)
        base image
```

Each group set to Pass Through unless it needs to be fenced. Each with a name that says what it does rather than what it is.

The free tools, honestly

Photopea implements Pass Through and Normal groups with Photoshop's semantics, and reads and writes them in PSD correctly. This is one of the places where it earns its reputation.

Krita has group layers with a *Pass Through* toggle that behaves the same way, plus a *Inherit Alpha* button which is Krita's version of clipping.

GIMP has layer groups, and its default is closer to isolated: a group composites internally and the mode applies to the result. GIMP added a pass-through mode in 2.10, and it is not the default, so a PSD built in Photoshop with adjustments reaching out of groups will not render identically in GIMP. If you exchange layered files with GIMP users, keep adjustments outside groups or flatten before handover.

Affinity Photo has groups with pass-through behaviour and, unusually, allows adjustment layers to be nested as children of a single layer without a group at all.

What this does not solve

Grouping is organisation, not isolation of *cost*. A group set to Normal has to composite its contents into a separate buffer before blending, which uses more memory than pass-through and can be slower on a large file. On an eight-gigabyte laptop with a 40-megapixel document, converting several big groups to Normal is a noticeable slowdown, and that is a real reason to reach for clipping instead where clipping will do.

And one structural limitation: a group can be masked and blended, but it cannot be referenced twice. If the same arrangement of layers needs to appear in two places, you are copying it, and the copies will drift. Smart objects — the next lesson — are the answer to that, and they bring their own costs.

BEFORE YOU MOVE ON

An editor puts four layers, including a Curves layer that was darkening the whole picture, into a group. To fade the group slightly they set its opacity to 80 per cent. The rest of the image immediately becomes lighter. What has happened?

A. The opacity change halved the Curves layer's own strength, so the darkening is now applied at 80 per cent everywhere.
B. Dropping below full opacity forces the group into isolated rendering, so the Curves layer now only affects the three layers inside it rather than the image below.
C. The Curves layer lost its mask when it was moved into the group, so it is no longer applying where it used to.
D. Group opacity is applied before the contents are composited, so each internal layer's effect is multiplied by 0.8 and the cumulative result is much weaker.

16 · 9 min

One file, forty mockups, one change

THE ONE THING TO KEEP

A smart object stores the original and re-renders it through your transform and filters each time, which is what makes scaling reversible, filters re-editable and contents swappable — at the cost of file size, render lag, and a portability gap that GIMP does not close.

A container, not a layer

Convert a layer to a smart object and the original pixels move inside a container. What you see on the canvas is a rendering of those contents at the current transform. The contents themselves are untouched.

That gives you three things an ordinary layer cannot do.

Non-destructive transforms. Scale a smart object to 20 per cent, work for an hour, scale it back to 100 per cent, and it is sharp. A normal layer scaled down has thrown away the pixels; scaling back up interpolates from what survived. The smart object re-renders from the stored original each time.

Smart filters. Any filter applied to a smart object becomes a re-editable entry beneath it, with its own mask and its own blend mode. Double-click the filter's name a week later and the dialogue opens with your settings. In GIMP and most free editors, a filter is applied and gone.

Contents you can replace. Right-click and *Replace Contents*, choose a different file, and everything — the transform, the smart filters, the mask, the warp — is re-applied to the new image.

The mockup workflow this enables

This is the practical payoff and it is worth doing once to see it.

You have a t-shirt photograph. You want to show a client's design on it. Put a flat rectangle where the print should sit, convert it to a smart object, warp it to follow the fabric's folds, set the layer to Multiply so the shirt's shading shows through, and mask it to the shirt's shape.

Now double-click the smart object's thumbnail. It opens as its own document. Paste the design in, save, and close. The mockup updates with the design warped, shaded and masked exactly as you set it up.

Forty designs become forty double-clicks rather than forty warps. Every downloadable mockup PSD you have ever seen works this way, and building your own takes twenty minutes.

Embedded against linked

A smart object can hold a **copy** of the source file inside the document, or a **link** to a file on disk.

Embedded is self-contained. Send the PSD and everything travels with it. The file is large, and two documents using the same logo hold two independent copies that will drift apart.

Linked — *Layer* > *Smart Objects* > *Convert to Linked* — stores a path. Update the logo on disk and every document using it updates. The file is small. And the link breaks the moment somebody moves the folder, renames the file, or sends you the PSD without the assets. Photoshop shows a warning badge and asks you to relocate it.

The rule that avoids grief: **linked for assets you control and reuse** — a logo, a price panel, a product cut-out used across a campaign — kept in a folder beside the document; **embedded for anything you are sending outside your machine**. *File* > *Package* collects a document and its linked assets into one folder, which is what you send.

What the free tools have

Photopea supports smart objects, including opening the contents, editing and updating, and it reads Photoshop's smart objects in a PSD. Smart filters are partially supported — some open as re-editable, others are baked. For mockups it works.

Krita has **file layers**: a layer that references an external image file and refreshes when that file changes on disk. That is exactly linked-smart-object behaviour, and Krita's version is arguably cleaner. It does not do smart filters.

Affinity Photo has embedded documents and live filter layers, which cover the same ground under different names.

GIMP has no equivalent. This is the most significant functional gap in GIMP for this kind of work, and it is honest to say so: scaling a layer down and back up in GIMP loses detail, and a filter applied is a filter baked. The workaround is to keep a duplicate of the original layer, hidden, at full size, and redo rather than re-edit.

The costs

Size. Every embedded smart object is a full copy of its source. A document with eight embedded 40-megapixel photographs is several gigabytes before you have made an edit.

Speed. The renderer re-renders the contents through the transform and every smart filter whenever anything changes. A smart object with three heavy filters on a large image makes every brushstroke lag.

Copies do not share. Duplicating a smart object with Ctrl/Cmd+J gives you a linked pair — edit one, both change. *New Smart Object via Copy* gives you an independent one. Two people have burned an afternoon on that distinction; know which you made.

Portability. A PSD full of smart objects and smart filters opens correctly in Photoshop and Photopea, approximately in Affinity, and poorly in GIMP. If the recipient's tool is unknown, ship a flattened TIFF alongside.

When to rasterise deliberately

Smart objects are not free and not always right. Rasterise when the layer is genuinely finished, when you need a pixel-level tool on it — the clone stamp will not paint on a smart object without rasterising — and when the file has become slow enough that the re-editability is costing more than it is worth.

The discipline is the same as everywhere else in this block: keep the version that still has the smart object, then rasterise in a copy.

BEFORE YOU MOVE ON

A designer builds a t-shirt mockup: a smart object warped to the fabric, set to Multiply, with a mask. They duplicate the layer with Ctrl/Cmd+J to make a second colourway, open the copy and replace its contents — and the first shirt changes too. What happened?

A. The mask is shared between duplicated layers, so editing either one propagates the change to both.
B. Multiply mode composites both layers against the same base, so any change to one is visible through the other.
C. Ctrl/Cmd+J duplicates the reference rather than the contents, so both layers point at the same stored source; New Smart Object via Copy makes an independent one.
D. The smart object was linked rather than embedded, so both layers are reading the same file from disk.

17 · 9 min

The panel where selections go to live

THE ONE THING TO KEEP

Alpha channels are permanent storage for selections, colour channels are ready-made masks waiting to be found, and multiplying two masks intersects them while screening them unions them.

Three colour channels, and as many alpha channels as you like

The Channels panel starts with the composite and the three colour channels. Below them you can store **alpha channels**: extra greyscale images, the same dimensions as the document, that are not displayed and carry no colour. They are storage.

Any selection can be written into one. *Select > Save Selection* in Photoshop and Photopea; GIMP calls the same thing *Select > Save to Channel*. Ctrl/Cmd-click the channel's thumbnail to load it back as a selection. That is the mechanism behind the most useful habit in this lesson: **a selection that took you twenty minutes should never be allowed to die when you press Ctrl+D.**

Alpha channels travel inside PSD and TIFF. They add about a third of a colour channel's worth of size each, which is negligible, and they are the reason a professional cut-out file can be re-cut six months later.

Building a mask out of a channel

The shortcut you will use most: Ctrl/Cmd-click the **RGB** composite thumbnail and you get a selection whose strength at every pixel equals that pixel's brightness — a luminosity selection with no edge anywhere. Ctrl/Cmd-click an individual channel and you get a selection based on that channel alone.

That second one is how difficult masks get made. Look at the three channels of a photograph of dark hair against a bright sky: in the blue channel the sky is near-white and the hair near-black, a far better separation than any selection tool will find. Duplicate the blue channel to a new alpha channel, apply a hard Levels move to push the sky to pure white and the hair to pure black, paint the middle of the head solid black with a brush, and you have a mask that took ninety seconds and holds every strand.

GIMP does the same through **Colours > Components > Decompose**, or by adding a layer mask with the *Grayscale copy of layer* option and then working on it. Krita has *Image > Separate Image*. None of this is a Photoshop-only technique.

Channel arithmetic: Apply Image and Calculations

Two dialogues that look forbidding and do something simple: they combine two channels with a blend mode and write the result somewhere.

Apply Image blends a chosen channel or layer into the currently active one. Point it at a layer mask and you can, for example, blend the blue channel into an existing mask with Multiply to knock back an area, in one step, with no brushwork.

Calculations takes two sources and writes the result to a new alpha channel, a selection or a new document. This is how multi-level luminosity masks are built, and it is the subject of a lesson of its own in the next block.

The practical pattern worth remembering: **Multiply two masks to get their intersection; Screen them to get their union.** Everything else in channel arithmetic is a variation on that.

```
intersection: result = maskA × maskB        (both must be white)
union:        result = 1 − (1−maskA)(1−maskB)  (either being white is enough)
```

Spot channels, for print

A **spot channel** is a different animal: it represents a specific named ink — a Pantone, a varnish, a white underbase for printing on dark garments, a die-cut line. It sits outside the CMYK separation and the printer produces a separate plate for it.

You will meet this if you do packaging, screen printing or anything with a metallic. The rules are strict: the channel must be named exactly as the printer expects, the ink must be specified, and overprint behaviour has to be agreed. Get it wrong and the job either prints as an extra unwanted plate or does not print at all. Ask the printer for their specification before building it; this is one area where guessing has a four-figure cost.

GIMP does not support spot channels. Scribus, free, handles spot colours properly for layout, and for a raster file destined for a spot plate the usual route is to supply a separate greyscale file per plate with clear naming.

Where channels mislead

Two honest cautions.

A channel-derived mask carries the channel's noise. In a high-ISO photograph the blue channel is the noisiest, so a mask built from it will have speckled edges. Apply a small blur to the alpha channel, or build from green instead and accept less separation.

Pushing Levels on a channel to make a mask destroys the partial values that make edges believable. Driving the sky to pure white and the subject to pure black gives you a crisp mask and no transition zone, which produces exactly the cut-with-scissors look the selections lesson warned about. The craft is to push hard in the flat regions and leave the transition zone alone, which usually means doing the Levels move, then painting the interior solid by hand rather than turning the contrast up further.

And a structural limit: alpha channels are per-document and per-file. They do not follow a layer when you copy it into another document, and they are silently discarded when you export to JPEG, PNG or WebP. The mask is a working asset. Save the PSD or TIFF, or it was never saved at all.

BEFORE YOU MOVE ON

To mask dark hair against a bright sky, an editor duplicates the blue channel and applies an aggressive Levels move until the sky is pure white and the hair is pure black everywhere, including the wispy strands. The resulting cut-out looks hard-edged and artificial. Why?

A. The blue channel is the noisiest, so the strands were destroyed by noise rather than by the Levels move.
B. Driving the transition zone to pure black and white removed the partial values that represent pixels genuinely mixed between hair and sky.
C. Levels cannot be used on an alpha channel, so the adjustment was silently applied to the composite instead.
D. A mask built from a single channel has no colour information, so it cannot represent a soft edge and needs to be rebuilt from the RGB composite.

18 · 8 min

The file you will need is the one from Tuesday

THE ONE THING TO KEEP

History dies when the document closes and a synced folder replicates your mistakes, so the record that actually rescues a job is named versions on disk plus three copies on two kinds of media with one of them elsewhere.

History is not a safety net

The History panel holds a list of recent states you can step back through. In Photoshop the default is 50; each state on a large document can cost tens of megabytes of scratch space, which is why the number is not higher.

Three things people assume about history that are wrong:

It is not saved with the file. Close the document and the history is gone. Open it tomorrow and there is one state.

It is linear by default. Step back five states, make a new edit, and the five you stepped over are destroyed. Photoshop offers *Allow Non-Linear History* in the History Options, which keeps them, at a cost in memory. It is off by default.

It fills up. Paint fifty brushstrokes and the state you actually wanted has been pushed off the end.

Snapshots partly fix this: click the camera icon and the current state is pinned, immune to the rolling window, until you close the document. Take one before any large irreversible operation. It still dies when the file closes.

GIMP's undo history behaves similarly and is bounded by a memory setting in Preferences. Krita's is also memory-bounded and, unusually, it can be configured to survive a crash through its autosave and recovery system.

What survives, and for how long

While the file is open	Fifty history states, linear by default. Step back five, make one edit, and those five are destroyed. Snapshots pin a state against the rolling window.
The moment you close it	History and snapshots are gone, including the snapshot you took before the irreversible operation. The document reopens with one state.
Every version you save	v01, v02, v03 on disk, zero-padded so v10 sorts after v02. Three a job: before anything destructive, before a client sees it, after they approve.
Up to about thirty days	A sync folder's own version history can still return Tuesday's file. It is a replica, not a backup: a deletion syncs, and so does a corruption.
For as long as you keep it	Three copies, two kinds of storage, one in another building. That covers the drive dying, the laptop being stolen, the flood, and deleting the wrong folder.

History is a convenience inside one session, not a record. The thing that actually rescues a job is a file on disk with a different name, and the thing that rescues the disk is a copy the live system cannot reach and overwrite.

Versions: the part that actually saves you

The reliable record is files on disk with different names.

A naming convention that works and sorts correctly:

```
honey-500ml_front_v01.psd
honey-500ml_front_v02.psd
honey-500ml_front_v03-client-notes.psd
honey-500ml_front_v03_APPROVED.psd
```

Zero-padded numbers so `v10` does not sort before `v2`. A short reason attached to versions that are not routine. The word APPROVED, or FINAL, used once and never followed by FINAL2 — if a revision comes after approval, it becomes v04.

Save a new version at each of these moments: before a destructive operation you cannot undo later, before sending anything to a client, and after a client approves. Three saves a job, and every one of them has a specific future use.

The habit that costs the most when missed: **save a version before flattening.** Flatten is the one command that turns a working file into a picture, and it is a single keystroke away from Save.

Backups: three copies, two media, one elsewhere

The standard rule, and it is standard because it survives the four realistic failures.

- **Three copies** of anything you cannot recreate.
- **Two different kinds of storage** — the laptop and an external drive, not two folders on the same disk.
- **One off site** — a different building, or a cloud account.

The four failures it covers: the drive dies, the laptop is stolen, the building floods, and you delete the wrong folder. Only the third is rare.

Free tools do this well. `rsync -a --delete source/ destination/` mirrors a folder to an external drive in one line and runs in seconds on subsequent passes because it copies only what changed. On Windows, `robocopy /MIR` is the equivalent. Both are built in.

The trap in cloud sync

A synced folder — Drive, Dropbox, OneDrive, iCloud — is **not a backup**. It is a replica. Delete a file, and the deletion syncs. Corrupt a file, and the corruption syncs. Ransomware encrypts a file, and the encryption syncs.

What sync services do provide is **version history**, usually 30 days, which genuinely recovers the Tuesday file. Learn where that is in the service you use before you need it, because finding it during a crisis is a bad time to learn. And know its limit: 30 days is not an archive, and files removed from the sync folder eventually leave the version history too.

A real backup is a copy the live system cannot reach and overwrite.

When a file will not open

Large layered files do occasionally corrupt — an interrupted save, a full disk, a flaky USB drive during a write.

- **Do not save over it.** Copy the broken file somewhere else and work on the copy.

- Try opening it in a different application. Photopea and GIMP both recover partially damaged PSDs that Photoshop refuses, because they are more tolerant of malformed sections.
- Photoshop writes a flattened composite inside every PSD for compatibility with other software. If *Maximise Compatibility* was on when it was saved, another application can often pull that composite out even when the layer data is unreadable. That setting costs file size and buys you this. Leave it on.
- Krita and GIMP both autosave recovery files; check the recovery prompt on next launch rather than dismissing it.

What none of this solves

Versions and backups protect the file. They do not protect the *decision*. Six versions in, nobody can remember why v03 looked better than v05, and comparing them means opening both.

The cheap fix is a plain text file beside the images — `notes.txt` — with two lines per version saying what changed and what the client said. It takes fifteen seconds and it is the only part of this lesson that survives you leaving the job.

BEFORE YOU MOVE ON

A designer keeps all working files in a cloud-synced folder and considers them backed up. On Thursday they accidentally flatten and save over a layered file, then close it. What is the realistic position?

A. The flattened save has synced everywhere, so the layered version is gone from every copy, but the service's file version history may still hold Wednesday's file for a limited window.

B. Sync services keep the most recent healthy version and reject a save that removes data, so the layered file is still the one stored remotely.

C. Reopening the file and using the History panel will step back past the flatten, because history is stored inside the document.

D. The layered data is still inside the file because a flatten only hides the layers, so unflattening restores them.

19 · 8 min

Eleven shortcuts and a scratch disk

THE ONE THING TO KEEP

Speed comes from never leaving the keyboard for the dozen commands you use hourly and from configuring scratch space and memory so the application stops stalling, not from working faster at the ones you use rarely.

Speed is not typing quickly

The difference between someone who edits for a living and someone who does it occasionally is not talent at the brush. It is that the professional never leaves the keyboard to change a brush size, never hunts a menu for a command they use forty times an hour, and has set the application up so it does not stall.

The shortcuts that pay for themselves in a week

Photoshop and Photopea share almost all of these; GIMP and Krita differ on some and can be remapped.

- **`[` and `]`** — brush smaller and larger. You will use this more than any other key.
- **Shift** + `[` / `]` — brush hardness down and up.
- `X` — swap foreground and background colour. Masking is painting black and white, so this is the mask toggle.
- `D` — reset to black and white.
- `B`, `E`, `S`, `J`, `P`, `L`, `W`, `C` — brush, eraser, clone stamp, healing, pen, lasso, quick selection, crop.
- `0`–`9` — set the current tool's opacity. `5` is 50 per cent, `55` typed quickly is 55.
- **Alt/Option + click** on a mask thumbnail — view the mask alone. `\` shows it as a red overlay.

- **Ctrl/Cmd + J** — duplicate the layer.
- **Ctrl/Cmd + Alt + Shift + E** — stamp visible onto a new layer.
- **Ctrl/Cmd + I** — invert. On a mask this is the single most-used command in local adjustment work.
- **Space bar held** — temporary hand tool, from inside any tool. Releases you from ever clicking the scroll bars.

Add **Ctrl/Cmd + Alt + Z** for step-backwards-through-history, which is different from plain undo in Photoshop and is the one people miss.

The method that makes these stick: pick three, force yourself to use only those for a day, then add three more. Trying to learn all eleven at once fails.

The settings that stop the stalling

Scratch disk. When the application runs out of memory it writes to disk. If that disk is the same nearly-full drive holding your system, everything crawls. *Preferences > Scratch Disks* in Photoshop; GIMP has a *Swap folder* in *Preferences > Folders*. Point it at the fastest drive with the most free space, and keep at least 20 GB free — more like 60 for large documents. The rule of thumb is that scratch use runs to several times the document size once you have a deep history.

Memory allocation. Photoshop defaults to about 70 per cent of RAM. On a machine where you also run a browser with thirty tabs, that fights. Setting it to 60 per cent often makes the whole system feel faster. GIMP's *Tile cache size* does the same job and defaults conservatively; raising it to half your RAM is usually a clear improvement.

History states. Lowering from 50 to 20 on a machine short of memory frees a surprising amount, at the cost of a shorter undo runway.

Turn off what you do not use. Photoshop's *Home Screen*, font previews on a machine with 900 fonts, and live thumbnail generation for large layer stacks all cost time.

Workspace and tool presets

Arrange the panels you actually use — Layers, Channels, History, Properties — and save the arrangement as a named workspace. On a small laptop screen, collapse everything else to icons and use Tab to hide all panels while judging an image.

Tool presets are underrated. A soft round brush at 30 per cent opacity and 0 per cent hardness, saved as "mask brush", is a click instead of three slider moves, forty times a day.

Tablets, and the honest case for and against

A graphics tablet gives pressure sensitivity: press harder, paint more opaquely or more widely. For masking, dodging and burning, and any retouching that involves building up an effect gradually, it is a genuine improvement and not a luxury. A small Wacom or Huion is £40 to £70 and lasts years.

What it does **not** improve: pen-tool paths, selections, crops, sliders, and anything geometric. A mouse is equal or better for all of those. Plenty of professional retouchers who work mostly on products use a mouse and do not feel disadvantaged.

If you do buy one, spend ten minutes on the pressure curve in the driver. The default is usually too firm, and adjusting it to reach full opacity with less pressure reduces the hand strain that otherwise builds up over a long day.

On that subject, and it belongs in a lesson about working fast: the injuries from this work are real. Wrist and shoulder pain from a badly positioned mouse, and eye strain from a bright screen in a dark room, both accumulate quietly. Screen at arm's length, top of the screen at eye level, elbow supported, and a break every 40 minutes. Nobody regrets this and plenty of people learn it the hard way.

The limit

A fast setup makes you quicker at what you already know how to do. It does not improve a decision. The editor who masks in half the time still has to decide whether the picture is better, and no shortcut helps with that. Spend the time you save looking at the image at fit-to-screen, which is the view your client will see it at.

BEFORE YOU MOVE ON

An editor's machine has 16 GB of RAM and a system drive with 8 GB free. Photoshop becomes unusably slow on a 40-layer document, though the Task Manager shows memory only two-thirds used. What is the most likely cause?

A. The layer count itself is the ceiling, and the only fix is to merge layers until it responds.

B. The memory allocation is set too low, so raising it to 100 per cent of RAM would resolve the stalling.

C. Scratch space on the nearly-full system drive has run out, so every operation is thrashing a disk with almost nothing free.

D. The history state count is at its default of 50, which is the single largest consumer of RAM in any document.

20 · 9 min

Working on a laptop that cannot hold the job

THE ONE THING TO KEEP

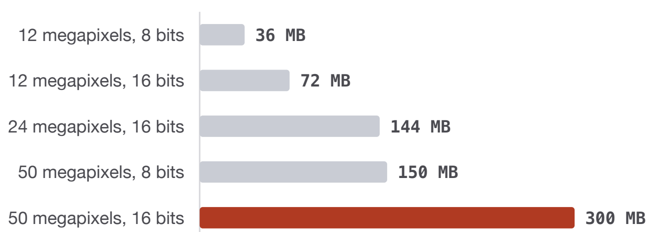
A slow document has four separable causes — pixel count, layer count, smart-filter re-rendering and scratch thrashing — and the fixes run from hiding groups and stamping checkpoints up to splitting the job into two documents, with browser-based tools hitting a hard memory ceiling rather than degrading gradually.

Four separate causes of a slow document

"It has got slow" is four different problems, and the fix differs for each. Diagnose before you start deleting work.

1. Pixel count. A 50-megapixel image in 16-bit is 300 MB per full-resolution layer before anything else. Every operation touches all of it.

One full-resolution layer, before anything else is open



Three channels, one or two bytes each, times the pixel count. Twenty adjustment layers re-composite all of that on every brushstroke. Photopea lives inside a browser tab with two to four gigabytes and no scratch disk, so it stops rather than slowing; GIMP and Krita swap to disk and degrade gradually.

2. Layer count and re-compositing. Change one layer and the application re-composites everything above it. Twenty adjustment layers means twenty operations per brushstroke.

3. Smart filters re-rendering. Each smart filter re-runs on every change to its smart object. A Gaussian blur at radius 200 on a large smart object is seconds of work, repeated constantly.

4. Scratch thrashing. The previous lesson's problem. Watch the efficiency indicator — Photoshop shows it at the bottom of the window if you set the status bar to *Efficiency*. Below 100 per cent means it is using disk; below 60 per cent means it is mostly using disk and you will feel it.

The five fixes, from cheapest to most drastic

Turn off what you are not looking at. Hide the groups you are not working on. Hidden layers still take memory but are not composited, which addresses cause 2 immediately.

Rasterise finished smart objects. Once the retouch inside a smart object is done and the transform settled, rasterising removes the re-render cost permanently. Save a version first.

Stamp a checkpoint. Ctrl/Cmd+Alt+Shift+E to flatten the visible result onto one layer, then hide everything below it and carry on above. The stack is preserved for a revision and costs nothing to composite while hidden. Name it with the date.

Split the job. Do the retouching in one document at full resolution and save a flattened TIFF. Do the compositing in a second document that places that TIFF. Two 400 MB documents open one at a time beat one 900 MB document open all the time.

Work at reduced size, apply at full. For work that is about placement rather than detail — a rough composite, deciding a crop, testing a grade — do it on a half-size copy. Curves, Levels, Hue/Saturation and most adjustment settings copy across to the full-size file unchanged, because they are functions of tone rather than of pixels. Masks and brushwork do not; those have to be done at full size.

Photopea's ceiling, stated plainly

Photopea runs in a browser tab, so it lives inside that tab's memory allocation. In practice this is around 2 to 4 GB depending on the browser and the machine, and there is no scratch disk to overflow into. When it runs out, the tab does not slow down gracefully — it stops or reloads.

What that means in practice: Photopea handles a 24-megapixel photograph with fifteen layers comfortably and will not handle a 100-megapixel poster with forty. Save often, and for genuinely large documents use a desktop application. This is not a criticism of Photopea; it is the architecture.

GIMP and Krita both have real tile-based memory management and swap to disk, so they degrade gradually instead. On a machine with 8 GB of RAM, GIMP on a large file will be slow and will keep working, which is often the more useful behaviour.

Working on low-end hardware, honestly

A large part of this readership is on a laptop with 4 to 8 GB of RAM, integrated graphics, and a nearly full drive. That is a real constraint and it is workable.

- **Downsize at the start, deliberately.** If the delivery is a 2,000-pixel web image, working on a 24-megapixel original is not producing quality, it is producing lag. Resize to 3,000 pixels on the long edge before you begin and you have ample margin.
- **8-bit unless the job needs 16.** Tonal stretching on a gradient needs 16; a product cut-out with modest adjustments does not.
- **Use GIMP or Krita rather than a browser tool** for anything large, for the memory reason above.
- **Close the browser.** Thirty tabs is two gigabytes.
- **An external SSD over USB 3** for scratch and files costs about £30 for 500 GB and is the single best value upgrade for this work. More RAM is better and is often soldered on modern laptops; a fast external drive is never soldered.

On a phone

Entirely viable for a real class of work, and worth knowing the limits. Snapseed keeps an editable stack and is free with no ads. Lightroom mobile's free tier develops raw and DNG parametrically, which is genuinely non-destructive. Photopea runs in a mobile browser and is painful to use on a small screen but works. Krita has an Android build.

What a phone does not do: precise masking at 100 per cent zoom, colour-managed print work, or anything with a deep layer stack. What it does well: developing a raw file, straightening, cropping, global tone and colour, and a quick cut-out. For a market trader photographing stock, that covers most of the job.

The limit worth accepting

Some jobs genuinely need hardware you do not have. A 200-megapixel architectural composite with sixty layers is not going to happen on an eight-gigabyte laptop, and no amount of technique changes that.

When you meet one, the answer is to reduce the scope of what is open at once — split the document, work in sections, flatten aggressively with versions saved — or to rent time on a better machine. What does not work is spending the day fighting an application and delivering late. Recognising the ceiling early is a professional skill, and telling the client on day one rather than day five is the part that matters.

BEFORE YOU MOVE ON

An editor working in Photopea on a 90-megapixel poster finds the tab reloads without warning after about twenty minutes, losing unsaved work. What is the correct reading of this?

A. The document's scratch file has filled the system drive, and freeing disk space will stop the reloads.
B. The internet connection is dropping, and Photopea reloads when it cannot reach its server.
C. The document has exceeded the browser tab's memory allocation, which has no disk overflow, so the tab is terminated rather than slowed.
D. Too many layers are set to a blend mode other than Normal, forcing isolated compositing that the browser cannot handle.

CASE STUDY · MODULE 2

The yearbook, the eight-gigabyte laptop, and the Tuesday crash

A government higher secondary school in Pune, where one teacher builds the yearbook between classes on a five-year-old laptop.

The job and the machine

Ninety-six pages, 340 student portraits, sixty event photographs and four full-bleed spreads. No Photoshop licence, so the teacher worked in Photopea, which runs in a browser tab, is free, and reads and writes layered PSD. She had used it for two previous yearbooks without trouble.

The trouble arrived with the class-of spread: forty cut-out portraits arranged on a painted gradient background, built at 3,000 pixels square in 16-bit because the gradient banded visibly in 8-bit when she stretched it. Each portrait was a smart object with a mask and a clipped Curves layer. At about the thirty-second portrait the tab reloaded, and two hours of arrangement went with it.

The mechanism is not mysterious. Photopea lives inside a browser tab's memory allocation, which is roughly two to four gigabytes depending on the browser and the machine, and there is no scratch disk to overflow into. A desktop application in the same position writes to disk and crawls. A browser tab in the same position stops.

The choice

Move to GIMP. Free, tile-based memory management, swaps to disk, so on an eight-gigabyte laptop a large file is slow and keeps working — which is usually the more useful behaviour. The cost is real: GIMP has no smart objects at all, so every scaled portrait is a resampled layer rather than a re-renderable one, and until version 3.0's layer effects it had no true adjustment layers either, so a tone move is baked into a duplicate. Three weeks before the printer's deadline, that is a new tool and forty placements to rebuild.

Stay and change the work. Split the document: portraits retouched in one file, the spread assembled in another from flattened TIFFs. Stamp a checkpoint and hide everything beneath it. Drop to 8-bit where the job does not need 16. Work at half size for placement and apply the settings at full. All of that is free and all of it costs her the re-editability she had been relying on.

Borrow the computer lab's newer machine. Sixteen gigabytes, available one hour a day between practicals, and the school's rules say files stay on the lab machine. One hour a day for a ninety-six-page book is not a plan, and a file that cannot leave the room cannot be worked on at home in the evening, which is when most of the yearbook was actually made.

What she weighed

The honest comparison was not GIMP against Photopea. It was *which parts of this job need which property*. The 340 portraits were reproduced at about 800 pixels on the page and needed a crop, a small exposure correction and a spot heal. Nothing there stretches a gradient, so nothing there needs 16-bit, and nothing there needs a smart object. The two gradient spreads needed 16-bit badly and needed almost no layers.

That observation split the job cleanly, and it is the part worth carrying away: the constraint was never the machine in general, it was one property of one part of the work.

> **STOP HERE**
>
> Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

She retouched the 340 portraits in GIMP at 8-bit, in batches of forty, with the heal work on an empty layer and *Sample merged* on. GIMP was slow and it never stopped. The two gradient spreads she built in Photopea at 16-bit with every other tab closed and nothing else in the document, stamping a checkpoint and hiding the stack beneath it every twenty minutes. The composite spread was assembled from flattened TIFFs rather than from live smart objects, which cost her the ability to swap a portrait without redoing its placement, and she swapped two.

The habit that came out of it was the file names. Every stage went to disk as `yearbook_spread-classof_v01.xcf` through `v09`, zero-padded, with a version saved before every flatten, copied to the school's Drive and to a pen drive she took home. Drive is a replica rather than a backup — a deletion syncs like everything else — but the pen drive lived in a different building, which was the point.

The book went to the printer four days early. The spread that had crashed was rebuilt in one evening the second time, because by then she knew which forty files she wanted and in what order.

Why did Photopea stop rather than slow down, and what would GIMP have done in the same position?

Photopea runs inside a browser tab and is limited to that tab's memory allocation, typically two to four gigabytes, with no scratch disk to spill into. When it runs out there is nowhere to go, so the tab reloads. GIMP and Krita have tile-based memory management and swap to disk, so the same document becomes slow and keeps working. On a laptop short of memory, degrading gradually is usually the more useful failure, which is why the choice of tool here is an architecture question rather than a quality one.

She kept 16-bit for two spreads and used 8-bit for 340 portraits. Defend that split.

Bit depth protects a tonal stretch. A smooth gradient stretched in 8-bit maps 256 levels onto a wider range and leaves gaps, and the gaps show as bands — which is exactly the painted background on the spread. The portraits reproduce at about 800 pixels, carry ordinary photographic tone that jumps from pixel to pixel anyway, and receive only a small exposure correction, so there is no stretch to protect and no smooth area to band. Working in 16-bit there would have doubled the file size and the memory for no visible gain, which on an eight-gigabyte laptop is a cost paid for nothing.

The school's Drive folder is not a backup. Why not, and what did the pen drive add?

A synced folder is a replica. Delete a file and the deletion syncs; corrupt one and the corruption syncs. What sync services do provide is a version history, usually about thirty days, which genuinely returns last Tuesday's file and is worth knowing how to reach before you need it. A real backup is a copy the live system cannot reach and overwrite. The pen drive was a second kind of storage kept in a different building, which covers the two failures Drive does not: the account or the folder being wrong, and the building itself. Three copies, two media, one elsewhere is the whole rule.

MODULE 2 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 You cut a product out last week by selecting the background and pressing Delete, then saved and closed. The client now wants a little of the original shadow back. What is the cheapest route?

A. Paint white on the mask with a soft brush
B. Step back through the History panel to before the delete
C. Add a drop shadow layer style to approximate the original one
D. Go back to the camera file and start again

2 A strong S-curve has made the skin look orange as well as more contrasty. What is the single-step fix?

A. Reduce the Curves layer's opacity until the colour settles down
B. Set the Curves layer's blend mode to Luminosity
C. Add a Hue/Saturation layer above it and lower the saturation
D. Apply the curve to the green channel only

3 A Curves layer that darkened the whole image stops reaching the layers below the moment you drag it into a group and set that group to 60 per cent opacity. What happened?

A. Group opacity multiplies with each layer mask inside the group and cancels out
B. Adjustment layers are designed to affect only their own group
C. Anything other than Pass Through isolates the group
D. The group inherited a clipping mask from the layer directly beneath it

4 Your clone and heal repairs sit at the top of the stack. You change the tone curve underneath and the repairs stop matching their surroundings. Why?

A. They were sampled from tones the curve has since moved
B. Clone layers cannot be re-rendered once the file has been saved
C. The clone tool always samples the flattened composite
D. Sample All Layers was switched off while you were painting them

5 An exported file is missing the last round of client changes, although those changes are plainly visible in the layered document. What is the likeliest cause?

A. Smart filters do not re-render until the file is reopened
B. The adjustment layers were set to Pass Through
C. The document was flattened in 8-bit instead of 16-bit
D. The export used a stamped layer made before the changes

6 A 40-layer, 60-megapixel document reloads the browser tab in Photopea and loses two hours. The same file in GIMP is slow and keeps working. What explains the difference?

A. Photopea works in 8-bit only, so it runs out of precision
B. Photopea has no scratch disk to overflow into
C. GIMP uses the graphics card and Photopea does not
D. GIMP compresses layers in memory while Photopea stores them raw

MODULE 3

Selections, cut-outs and edges

Getting a subject out of its background is the most requested skill in this trade and the one where the difference between adequate and professional is visible at a glance. This block goes past the automatic button: bezier paths for manufactured edges, painting a selection by hand, colour and luminosity as selection sources, what the segmentation models are actually doing, the equation that makes hair mathematically hard, and how to cut out things that are not opaque at all.

> BY THE END OF THIS MODULE YOU CAN
>
> Choose and execute the right extraction method for a given edge — path, painted mask, colour range, luminosity, model or blend-mode composite — decontaminate the result so it carries no trace of its old background, and run a batch of cut-outs to a consistent standard with a sampling check that catches the failures

21 · 10 min

Selections are easy until you reach hair

THE ONE THING TO KEEP

Edge pixels are mixtures of subject and background, so a good cut-out is about partial transparency and colour decontamination, not about finding a line.

A selection is a number per pixel, not a line

The marching-ants outline lies to you. It draws a boundary, but underneath, a selection is exactly what a mask is: a value from 0 to 255 for every pixel saying how selected it is. That is why selections and masks convert into each other freely, and why "the selection is off by one pixel" is usually the wrong diagnosis.

Choose the tool by the kind of edge you are dealing with.

The four tools and when each is right

Magic Wand — click a pixel, and it takes every connected pixel within a colour tolerance of it. Tolerance 32 by default; raise it for a gradient sky, lower it for a busy one. Only sane on flat, even backgrounds. Turn *Contiguous* off and it selects that colour everywhere in the frame.

Quick Selection / Object Selection — you drag, and the tool grows the selection outward, stopping where it detects an edge. Object Selection and *Select Subject* go further and use a trained segmentation model to find a person or product in one click. This is the everyday tool and it is genuinely good. Photopea has a quick-selection tool and an automatic subject cut-out; GIMP has the *Foreground Select* tool, which works on the same rough-then-refine principle and takes longer to get to the same place.

Colour Range (*Select > Colour Range*) — samples a colour and selects by distance from it, with a Fuzziness slider and a live black-and-white preview. Better than the wand because it is not limited to a connected region and gives you partial selection rather than in-or-out. It is the right tool for a red dress across a whole frame, or for a sky through tree branches.

The Pen tool — you draw a vector path by hand, then convert it to a selection. Slow, and unbeatable for manufactured objects with hard edges: a bottle, a phone, a shoe, a box. Because it is a mathematical curve rather than a pixel guess, the edge is smooth at any zoom, and every e-commerce cut-out you have seen that looks expensive was done this way. Inkscape logic applies here too — see vector and print. Photopea and GIMP both have a pen/paths tool with the same behaviour and roughly the same shortcuts.

Two habits worth building: hold **Shift** to add to a selection and **Alt/Option** to subtract, and select the *simpler* region then invert. Selecting a plain wall and inverting is far quicker than tracing a person.

Hair, and why refine-edge exists

Here is where beginners lose an afternoon.

At the edge of hair, a single pixel is not hair and it is not background. The lens and sensor averaged both into one value. That pixel is genuinely, physically, 40 per cent hair and 60 per cent wall. No boundary exists to find, so any hard selection is a lie in one direction or the other.

The answer is partial alpha, and that is what **Select and Mask** (Photoshop; earlier called Refine Edge) is for. The important control is the **Refine Edge brush** — you scrub roughly over the transition zone and the tool estimates, per pixel, what fraction of it belongs to the subject.

The second control is the one people miss: **Decontaminate Colours**. Consider a person photographed against a green wall. The semi-transparent hair pixels literally contain green — the wall's colour is baked into their values. Place them on white and you get a green fringe. Contracting the selection does not fix this; it just eats the hair. Feathering does not fix it either; it smears the green. Decontaminate Colours rewrites the colour of those partial pixels to match the subject while keeping their transparency. It is the only control that addresses the actual cause.

This is also why professionals shoot cut-outs against grey or white rather than a colour, and why a green screen is used for video and rarely for stills.

The free paths, honestly

- **Photopea's** automatic cut-out handles most product and person shots to a standard that passes on a marketplace listing. Its hair refinement is weaker than Photoshop's.
- **remove.bg** and similar web services do one-click background removal. The free tier gives you a small image; the full-resolution file is paid. Also: you are uploading a client's photo to somebody's server, which is a decision, not a convenience.
- **rembg** is the open-source version of the same idea. It runs on your own machine, on the CPU if you have no GPU, and it costs nothing per image. Slower, private, unlimited.
- **On a phone**, most recent Android and iOS galleries let you long-press a subject and lift it out. The alpha is rough at hair but perfectly good for a social post.
- **GIMP** with Foreground Select plus hand-painting on the mask will get you anywhere Photoshop will. It will take three times as long on hair.

How to spot a bad cut-out

Look for these before delivering, at 100 per cent zoom:

- A pale or dark **halo** tracing the whole outline — the selection included a rim of the old background.
- An edge that is **sharper than the rest of the photo**. If the subject was shot at f/2 and slightly soft, a razor edge reads as fake instantly.
- **No contact shadow.** An object with a perfect edge and nothing under it floats. Keep the real shadow or rebuild one.
- Hair that has turned into a **helmet** — the fine strands were all sacrificed to get a clean line.

Today: cut one thing out with Quick Selection and the same thing with the Pen tool. Time both. You will know from then on which job deserves which.

BEFORE YOU MOVE ON

A portrait shot against a green wall is cut out and placed on white. A thin green edge shows around the hair. What is actually going on, and what fixes it?

A. The selection is one pixel too generous, so contracting it by one pixel before masking removes the green.
B. The green channel is clipped in the original exposure, so the file needs re-shooting.
C. The mask needs more feathering so the transition is softer and the green stops being noticeable.
D. The semi-transparent hair pixels are physically part hair and part wall, so they contain green; Decontaminate Colours rewrites those pixel colours while keeping their transparency.

---

22 · 10 min

Two handles and a curve: making the pen tool obey

THE ONE THING TO KEEP

A bezier path is anchors plus direction handles, so accuracy comes from placing few anchors at direction changes and shaping the curve with handles — and the finished cut-out lives or dies on a sub-pixel feather that matches the photograph's own edge softness.

Why it feels hostile

The pen tool is the one that makes people close the application. It behaves unlike every other tool: clicking does not draw, dragging does something unexpected, and the line jumps around after you thought you had finished it.

It is worth twenty minutes because for any manufactured object — a bottle, a phone, a box, a shoe, a car — a path gives an edge that no pixel-based tool can match. The curve is mathematical, so it is exactly as smooth at 400 per cent zoom as at fit-to-screen, and it can be nudged a week later without redoing anything.

What a bezier curve is

A path is a chain of **anchor points**. Between any two anchors, the curve's shape is controlled by **direction handles** sticking out from them. The handle's angle sets the direction the curve leaves the anchor in; the handle's length sets how strongly it pulls before the next anchor takes over.

That is the whole model. Everything else is consequences.

- **Click** to place an anchor with no handles: a corner.
- **Click and drag** to place an anchor and pull out a pair of handles: a smooth point, where the curve passes straight through.
- The direction you drag is the direction the curve will leave in. Drag in the direction of travel, not away from it.
- **Alt/Option-drag a handle** to break the pair, so the curve can arrive smoothly and leave at an angle. This is how you make a rounded corner meet a straight edge.

The rule that halves your anchor count

Beginners place forty anchors round a bottle. Professionals place twelve. The rule:

> *Put an anchor where the curve changes direction, and nowhere else.*

A bottle's silhouette has a shoulder, a waist, a base corner. Each of those is a direction change and deserves an anchor. The smooth run between them does not; that is what handles are for. Extra anchors do not add accuracy — they add wobble, because each one is a place where your hand can be slightly wrong.

A second rule that helps: place anchors at the **extremes**, the topmost, bottom-most, leftmost and rightmost points of each curve, with handles horizontal or vertical at those points. Hold Shift while dragging to constrain a handle to 45-degree increments and the curve snaps into a clean shape.

Editing after the fact

The path is not committed until you decide it is.

- **Direct Selection tool (A)** moves anchors and handles.
- **Ctrl/Cmd held** while in the pen tool temporarily gives you the Direct Selection tool, so you can adjust without switching.
- **Alt/Option held** over an anchor converts between smooth and corner.
- Clicking on a path segment with the pen adds an anchor; clicking on an existing anchor removes it.

Paths live in the Paths panel and are saved inside the PSD or TIFF. Double-click the temporary "Work Path" to name and keep it, or the next path you draw replaces it. That one habit prevents the commonest loss in this lesson.

Turning a path into a cut-out

The Paths panel's buttons: load as selection (or Ctrl/Cmd + click the path thumbnail), fill, stroke.

The practical sequence for a product:

1. Trace the silhouette. Sit the path **half a pixel inside** the edge, not on it, so no rim of old background survives.
2. Load as a selection.
3. Feather 0.3 to 0.5 pixels — enough that the edge is not aliased, not enough to look soft.
4. Click **Add Layer Mask**, never Delete.

The feather number matters more than it looks. Zero gives a jagged, cut-with-scissors edge that reads as fake instantly. Two pixels gives a soft glow that reads as fake for the opposite reason. On a normally-shot photograph, the subject's real edge is about half a pixel soft, and matching that is the whole trick.

From path to cut-out, and the number that decides it

Anchor only where the curve changes direction	Sit the path half a pixel inside the edge	Load the path as a selection	Feather 0.3 to 0.5 pixels	Add a layer mask, never Delete
A bottle has a shoulder, a waist and a base corner. Twelve anchors, not forty: every extra one is another place your hand can be slightly wrong.	On the edge leaves a rim of the old background all the way round, and that rim is what reads as a halo once the subject is on white.	Name the path in the Paths panel first, or the next path you draw replaces the temporary Work Path and the tracing is gone.	Zero reads as cut with scissors. Two pixels reads as a glow. A normally shot photographic edge is about half a pixel soft, and matching it is the whole trick.	The pixels stay, so next week's revision is a brushstroke. The path stays too, inside the PSD or TIFF, so the cut can be redone in six months.

Every e-commerce cut-out that looks expensive was made this way. The pen tool is the wrong choice for hair, fur, foliage against sky or anything out of focus, because a path draws a boundary and those subjects do not have one.

Clipping paths, for print

A path stored in a TIFF or EPS and designated as a **clipping path** tells layout software the outline to flow text around and the shape to knock out. Printers and magazine designers still ask for this, and supplying a PNG instead is not the same thing.

In Photoshop: name the path, then *Paths panel menu > Clipping Path*, and set the flatness (a device tolerance — leave it blank unless the printer specifies). Save as TIFF or EPS. GIMP can export paths as SVG but does not write Photoshop clipping paths, which is a genuine gap if you do print production; Photopea does write them.

Practising in the right place

The pen tool in Inkscape, Illustrator, Photoshop, Photopea, GIMP, Figma and Affinity is the same tool with the same handles and nearly the same modifier keys. Learn it once and it transfers everywhere, which makes it one of the highest-return half-hours in this whole course.

A good practice exercise, and it is genuinely how people learn it: open a photograph of a wine bottle at 200 per cent and trace it, aiming for under fifteen anchors. Do it again tomorrow. By the fourth attempt it will stop feeling hostile.

When not to use it

The pen tool is wrong for anything with a soft, fibrous or transparent edge. Hair, fur, foliage against sky, smoke, out-of-focus flowers — a path draws a hard boundary, and those subjects do not have one. Forcing a path onto hair is the technique that produces the helmet look.

It is also slow. Tracing a complex machine can take forty minutes. When the subject is a person in a soft-focus photograph, the automatic tools plus a painted mask will be faster and better. The skill is not using the pen tool for everything; it is recognising the twenty per cent of jobs where nothing else gives that edge.

BEFORE YOU MOVE ON

A retoucher traces a phone with the pen tool using forty anchor points placed evenly around the outline. The resulting cut-out edge looks subtly wavy along what should be dead-straight sides. Why did more anchors make it worse?

A. Each anchor introduces a place where the hand's position and the handle angle can be slightly off, so a straight run described by many anchors accumulates small errors a two-anchor segment cannot have.
B. Photoshop smooths paths with fewer than twenty anchors automatically, and the smoothing is disabled above that count.
C. Dense anchors force the path to be rendered as a polyline rather than curves, which produces visible faceting.
D. The feather value was too low for a path with many anchors, and raising it to two pixels would hide the waviness.

23 · 8 min

Stop fighting the selection tools and paint it

THE ONE THING TO KEEP

Quick Mask turns a selection into a paintable greyscale image, so the fast route to a difficult cut-out is an automatic tool for the bulk followed by a low-opacity brush for the part the tool got wrong.

The mode that makes a selection into an image

Press **Q** and the marching ants disappear, replaced by a red overlay. You are now in Quick Mask mode, and the selection is being displayed as a temporary greyscale image: selected areas clear, unselected areas red, partially selected areas partly red.

While you are in that mode, **every painting tool works on the selection.** Paint with black and the red spreads, removing area from the selection. Paint with white and the red is wiped away, adding area. Grey adds partial selection. Use a gradient, a blur, a texture brush — anything that makes marks makes selection.

Press **Q** again and you are back to a normal selection, ready to become a mask.

GIMP has the identical feature under *Select > Toggle Quick Mask* or **Shift+Q**, with the same red overlay. Krita has *Selection Masks*, which are similar and persist. Photopea has Quick Mask on the same Q shortcut.

Why this changes how the work feels

The selection tools are all attempts to guess a boundary. When the guess is good, they are fast. When the guess is bad — a low-contrast edge, a busy background, an object that shares colour with what is behind it — you can spend twenty minutes fighting tolerance sliders and add-and-subtract clicks.

Quick Mask removes the guessing. The rough selection from Quick Selection gets you eighty per cent of the way in two seconds; Q, then eight brushstrokes, gets you the rest. This is faster than any amount of tool-tuning, and it produces a better edge, because you are making decisions rather than adjusting a threshold that applies everywhere at once.

The two settings people never find

Double-click the Quick Mask button at the bottom of the toolbar and a dialogue appears with two choices worth knowing.

Colour Indicates: Masked Areas or Selected Areas. The default is masked, meaning red is what is *not* selected. If your painting seems inverted, this is why. Flip it to match how you think.

Opacity and colour of the overlay. The default is 50 per cent red. When you are masking something red, that is useless. Change it to green or blue for that job. This takes five seconds and saves a lot of squinting.

The brush settings that do the work

Masking is painting, so the brush settings are the technique.

- **Hardness.** A hard edge (100 per cent) for a manufactured object. Soft (0 to 20 per cent) for shadows, hair, out-of-focus regions. A soft brush is a feather you control per stroke.
- **Opacity at 20 to 40 per cent**, built up in passes. One stroke at full opacity is a decision you cannot modulate; four at 25 per cent lets you stop where it looks right.
- **Flow** is different from opacity: opacity caps the maximum a single stroke reaches, flow controls how quickly each dab builds towards it. Low flow with high opacity is the airbrush feel, best for shadow work.
- **Size with `[` and `]`.** Constantly. A mask painted with one brush size is always visible as a mask.

Refining an existing mask with paint

The same idea applies to a mask that already exists, without Quick Mask at all: click the mask thumbnail and paint on it directly. Alt/Option-click it first to see the mask on its own in black and white, which is the only way to find the ragged edges you cannot see against the picture.

Three repairs that come up constantly:

A halo of old background. Paint black along the edge with a small soft brush at low opacity, following the outline. Two passes usually removes it.

A hole in the middle. Paint white with a large hard brush. The interior of a subject has no edge to preserve, so precision there is wasted effort.

A too-hard transition. Select the mask, apply a Gaussian blur of 0.5 to 1 pixel, then a Levels move on the mask to pull the blurred edge back towards where it was. Blurring alone moves the edge outward slightly; the Levels correction is what keeps it in place.

Making a mask out of something that is not a selection

Because a mask is just a greyscale image, anything greyscale can become one. Copy a photograph, paste it into a mask, and the photograph's tones now control visibility. Paste a scanned ink texture in and you get a grungy torn-edge effect with no brushwork at all. Paste a gradient and you have a graduated filter.

Alt/Option-click a mask thumbnail to enter it, then Ctrl/Cmd+V. This is the move behind most "how did they do that edge" questions.

The honest limits

Painting a selection is precise and it is **slow**. A complicated cut-out done entirely by hand at 100 per cent zoom is an hour. Use the automatic tools for the bulk and paint the fifteen per cent they got wrong.

It is also **resolution-dependent** in a way that catches people out. A mask painted while zoomed to 200 per cent can look crude at fit-to-screen, and a mask that looks perfect at fit-to-screen can be visibly blocky at 100 per cent. Check at both, and check the finished composite at the size it will be seen.

And hand-painting cannot create partial transparency that reflects reality. You can paint a 50 per cent grey along a hair edge, and it is your estimate of the coverage rather than a measurement of it. For genuinely fine edges, the matting methods later in this block do better, because they solve for the coverage from the pixel values instead of guessing it.

BEFORE YOU MOVE ON

An editor enters Quick Mask mode to refine a selection around a red jacket and finds they cannot see what they are doing. What is the correct fix?

A. Switch Colour Indicates from Masked Areas to Selected Areas, which reverses which part of the image is covered.

B. Leave Quick Mask and refine with the lasso instead, since Quick Mask is unsuitable for saturated subjects.

C. Reduce the overlay opacity to 20 per cent so the jacket shows through more clearly.

D. Double-click the Quick Mask button and change the overlay colour from red to green or blue for this job.

24 · 8 min

Tolerance is a wall, fuzziness is a slope

THE ONE THING TO KEEP

The Magic Wand's tolerance produces an in-or-out selection while Colour Range's fuzziness produces partial values, which is why one is for flat graphics and the other for anything that shades — and neither can separate a subject from a background that shares its values.

Two colour-based tools that are not the same tool

The Magic Wand and Colour Range both select by colour, and people use them interchangeably. They compute different things.

Magic Wand with Tolerance 32 asks, of each pixel: is this within 32 units of the sampled colour? Yes or no. The answer is binary. Every pixel is either fully selected or not selected, and the boundary between them is a step. The only softening you get is the anti-aliasing checkbox and whatever feather you apply afterwards.

Colour Range with Fuzziness 32 asks: how close is this pixel to the sampled colour? It returns a **partial** selection — a pixel slightly outside the core range is selected at 60 per cent, one further out at 20 per cent. The result has a gradient built in, which is why it produces far better masks on anything that shades.

That is the practical difference, and it is why Colour Range is the better tool for almost everything except a flat screenshot.

Tolerance is a wall, fuzziness is a slope

One answers yes or no, so the boundary is a step and the only softness available is the feather you apply afterwards. The other returns partial values, which is what a sky seen through tree branches needs. Neither separates a brown dog from a brown sofa: the numbers genuinely overlap, and no setting invents a difference.

GIMP's equivalents: *Fuzzy Select* (the wand) has a Threshold and a **Draw mask** preview; *Select by Colour* does the same thing non-contiguously. GIMP's threshold is also essentially binary, and its soft equivalent is to build the mask from a decomposed channel instead.

The controls in Colour Range that matter

Select > Colour Range in Photoshop and Photopea.

Shift-click to add more samples. One click on a sky selects one blue. A sky runs from pale at the horizon to deep overhead; three or four shift-clicks across it produce a selection that covers all of them. This single habit doubles the tool's usefulness.

Localized Colour Clusters, with its Range slider, restricts the selection to pixels *near* where you sampled. Without it, sampling a red dress also selects a red sign on the far side of the frame. With it, you can pick one of two identical red objects.

The preview dropdown. Set it to Greyscale and the dialogue shows the mask you are about to make, in black and white, live. Judge it there, not by the marching ants afterwards.

The preset list includes Reds, Yellows, Highlights, Midtones, Shadows, Out of Gamut and **Skin Tones**. Skin Tones with *Detect Faces* on is a surprisingly good starting mask for portrait work — not because it is accurate, but because it is 80 per cent right in two seconds and paintable from there.

Where colour selection is exactly right

- **A sky through tree branches.** No selection tool traces a thousand twigs. Colour Range on the blue does it instantly, and the partial values along each branch are approximately correct.
- **Changing one colour across a whole frame** — every red item in a catalogue shot, a brand colour on signage.
- **Selecting out-of-gamut pixels** before a print conversion, to see what is going to clip.
- **Building a starting mask for a luminosity or colour-specific adjustment**, where you will paint the rest.

Where it fails, and the mechanism

Colour selection fails when subject and background **share colour values**, and no setting fixes that, because the tool has no other information. A brown dog on a brown sofa, a grey car against a grey wall, a white shirt against a white wall — the numbers genuinely overlap.

The diagnosis is quick: look at the three channels individually. If one of them separates subject from background, build from that channel (the previous block's technique). If none of them does, colour is not the route and you need geometry — a path, a painted mask, or a model that understands objects rather than values.

The second failure is **noise and compression**. A JPEG's 8×8 blocks and a high-ISO file's speckle both mean that visually uniform areas are numerically scattered, so a colour selection comes out with holes and a ragged edge. Applying a one-pixel blur to the image on a duplicate layer before sampling, or blurring the resulting mask, both help; neither is free of cost.

The free path

GIMP's Fuzzy Select has a genuinely useful feature Photoshop lacks: dragging after clicking adjusts the threshold live, with the mask previewed. It is a faster way to find the right value than typing numbers.

For partial, gradient-aware colour selection, GIMP's route is **Colours > Components > Decompose** to HSV, then work on the hue or saturation layer as a mask. More steps, same result, and it teaches you what the tool is doing.

Krita has *Select Similar Colour* and a *Colour Range* selection, both adequate.

One caution about recolouring

Colour Range is often the first step in changing an object's colour. A warning that belongs here rather than later: a colour selection selects pixels that *are* a colour, including the highlights and shadows on the object, which are the same hue at different lightness. Recolour the selection with a Hue shift and the shading usually survives. Recolour it by painting a flat colour and the object goes dead, because you have replaced its shading with a single tone.

The method that keeps the shading: a Hue/Saturation layer with the selection as its mask, adjusting Hue rather than painting, or a solid colour layer set to **Colour** blend mode, which takes the hue from the fill and the luminosity from the photograph. That second one is the standard trick and it takes ten seconds.

And the honest limit on all of it: changing a red car to yellow works; changing a black car to yellow does not, because black has no hue to shift and no shading information in the colour channels to preserve. Dark and desaturated objects have to be repainted with the tone rebuilt by hand, which is a job of a different size.

BEFORE YOU MOVE ON

An editor uses Colour Range to select a black leather bag so they can recolour it tan, and finds that neither a Hue shift nor a Colour-mode fill produces a convincing tan bag. Why?

A. Colour Range cannot sample dark values, so the selection itself is empty and nothing is being recoloured.
B. Black pixels carry almost no hue or saturation to shift, and the shading that would define a tan surface has to be rebuilt by painting rather than remapped.
C. Colour blend mode takes luminosity from the layer below, and leather has too much specular highlight for that mode to work on any colour.
D. The bag needs to be selected with Localized Colour Clusters enabled, because without it the selection includes other dark areas and dilutes the adjustment.

25 · 9 min

Masks with no edge anywhere

THE ONE THING TO KEEP

A luminosity mask makes each pixel's participation equal to its own brightness, so an adjustment fades continuously and has no boundary to halo along — and intersecting the mask with itself narrows it to the highlights, the shadows or a midtone band.

Selecting brightness instead of regions

Every selection made so far has drawn a boundary somewhere. A luminosity mask does not. It selects by tone: each pixel's participation equals its own brightness, so the mask varies continuously across the whole frame and there is no line for a halo to form along.

The basic one is two clicks. Open the Channels panel and **Ctrl/Cmd-click the RGB composite thumbnail**. You now have a selection at 100 per cent in the white areas, 50 per cent in mid-grey, zero in black. Add any adjustment layer and it arrives with that as its mask.

That single move is the correct way to recover a bright sky without the halo that a hard selection produces along a roofline. It is also the correct way to add contrast to the shadows without touching the highlights, by inverting it first.

Building the levels

One luminosity mask is blunt. The standard toolkit has several, each narrower than the last, and they are built by intersecting a mask with itself.

The arithmetic from the channels lesson: **multiplying two masks intersects them.** Intersect the basic brightness mask with itself and the result is brightness squared — a mask that is near-zero through the midtones and only opens up in the genuinely bright areas. Do it again and it is narrower still.

The traditional naming, from the photographer Tony Kuyper who formalised this:

- **Lights 1** — the plain luminosity mask.
- **Lights 2** — Lights 1 intersected with itself. Only the brighter half of the highlights.
- **Lights 3** — narrower again.
- **Darks 1, 2, 3** — the same, built from the inverted mask.
- **Midtones** — the full mask minus both the lights and the darks, giving a band that peaks in the middle and falls away at both ends.

In practice: load the luminosity selection, then hold **Ctrl/Cmd + Alt/Option + Shift** and click the RGB thumbnail again. That modifier combination intersects the new selection with the existing one. Repeat for a narrower band. Save each one into an alpha channel as you go, because you will want them again in the same document.

Calculations (*Image > Calculations*) does the same thing in one dialogue: two sources, Multiply, result to a new channel. It is faster once you know what you are asking for.

What this is actually for

Recovering a sky. A Darks mask on a Curves layer lets you lift the shadows in a landscape without the sky going grey — the adjustment simply does not reach the bright pixels, and there is no boundary between treated and untreated regions.

Contrast that does not clip. An S-curve applied through a Midtones mask steepens the middle and leaves the ends alone, which is the thing the tone lesson said contrast normally borrows from.

Colour in one tonal band. A warm colour layer masked to Lights and a cool one masked to Darks is split toning, done properly, with control over exactly how far each reaches.

Dodging and burning that respects the subject. Burning through a Lights mask darkens only what is already bright, so a highlight is brought down without muddying the shadow beside it.

Sharpening the right part. A High Pass layer masked to Midtones sharpens texture without exaggerating noise in the shadows or halos in the highlights.

The free routes

GIMP builds these natively. Add a layer mask with the *Grayscale copy of layer* option and you have Lights 1 immediately. For narrower bands, duplicate the mask into a channel and use *Colours* > *Curves* on it, or use the layer mode arithmetic. GIMP's channel arithmetic through *Filters* > *Generic* > ... and the GEGL operations covers the rest.

Photopea has the Channels panel with Ctrl-click loading and the same intersect modifiers, so the whole technique transfers directly.

Krita has *Filter Masks* and can copy a layer into a mask, which achieves the same thing with a different route.

Paid panels exist that automate all of this — they generate sixteen masks and give you buttons. They save clicks and teach you nothing. Build them by hand five times and you will know which one you need before you make it, which is faster than scrolling a panel of sixteen.

Where they fall down

They ignore what the picture contains. A luminosity mask selects a white shirt and a white cloud identically, because it only knows brightness. If you want the sky and not the shirt, you need to combine the luminosity mask with a region mask — load the luminosity selection, then paint black over the shirt on the resulting mask. Most real uses are this combination rather than the luminosity mask alone.

They are tone-dependent, so they move when the tone moves. Build a Lights mask, then change the exposure underneath it, and the mask no longer corresponds to the highlights. Masks are snapshots. Rebuild them after any large tonal change, or apply tonal changes above them.

Noise gets selected too. In shadows, the luminance variation that is noise becomes variation in the mask, and an adjustment applied through it can amplify the noise pattern. Blurring the mask by one or two pixels fixes this at negligible cost to the effect.

Banding, in 8-bit. A continuously varying mask in an 8-bit file has 256 steps like everything else, and a strong adjustment applied through it can show them as terracing in a smooth sky. This is the clearest everyday reason to work in 16-bit.

BEFORE YOU MOVE ON

A landscape photographer uses a Lights mask to bring down a bright sky, and it works well. In the same frame, a white building in the foreground also darkens, which they did not want. What is the correct reading?

A. The mask needs narrowing to Lights 3, which would exclude the building while keeping the sky.
B. A luminosity mask ranks pixels only by brightness, so it cannot distinguish two bright objects, and the fix is to paint the building out of the resulting mask.
C. The mask was built from the RGB composite rather than the blue channel, which would have separated sky from building.
D. The adjustment layer is above the building in the stack, so it is reaching content it should not; moving it lower would exclude the foreground.

26 · 9 min

What the one-click button is actually doing

THE ONE THING TO KEEP

A one-click cut-out is an object-recognition model predicting which pixels belong to a learned shape, which is why it fills enclosed gaps, loses thin structures and guesses at transparency — and why the mask must be inspected on its own before it is used.

Not an edge detector

Select Subject, Remove Background, the long-press-and-lift on your phone: all of them run a neural network that was trained on hundreds of thousands of images paired with human-drawn masks. The model has learned what a person, a car, a dog, a bottle look like as shapes. It is not measuring contrast at a boundary; it is recognising an object and predicting which pixels belong to it.

That difference explains both why it works where the wand fails and how it fails.

The common architectures, in the free world:

- **U²-Net** — the model behind **rembg** and a great many web background removers. Small, fast, runs on a CPU in a second or two, trained for salient object detection: it finds the *one thing the picture is about*.
- **Segment Anything (SAM)**, from Meta, released free with open weights in 2023. Different job: you give it a click or a box and it returns a mask for whatever is there. It will segment anything, including things nobody labelled, which is why it is the basis of many newer tools.
- **MODNet** and **BiRefNet** — portrait and high-resolution matting models, better on hair than U²-Net.

Running them free, locally

```
pip install rembg
rembg i input.jpg output.png          # one image
rembg p ./in ./out                     # a whole folder
rembg i -m isnet-general-use in.jpg out.png   # a different model
```

That runs on your own machine, on a CPU, with no upload and no per-image cost. On a five-year-old laptop it takes two to five seconds per image. There is a `rembg[gpu]` variant if you have one.

There are also GIMP plugins that wrap the same models, and a growing number of offline Android apps.

This matters for a specific reason beyond money. Using a web background remover means uploading a client's photograph to somebody else's server, under terms of service you have not read, which may permit training on it. For a product shot of a jar of honey nobody cares. For an unreleased product, a medical image, a photograph of a child, or anything under a confidentiality agreement, it is a decision you should be making deliberately rather than by habit.

Where models fail, and why each failure happens

Two objects, one picked. Salient-object models return *the* subject. Two people standing apart and it may take one. SAM-style click-based tools do not have this problem, because you tell it what you mean.

Holes get filled. A mask predicted from learned object shapes tends to produce closed, plausible silhouettes. The gap between an arm and a torso, the space inside a handle, the mesh of a bicycle wheel — these are frequently filled in, because most training examples had solid bodies there. Always check the enclosed gaps.

Thin structures disappear. Guitar strings, spectacle arms, antennae, a dog's whiskers. These occupy a few pixels and contribute almost nothing to the loss function the model was trained against, so the model is not penalised much for missing them.

Reflective and transparent objects are guessed. A wine glass gets a solid silhouette rather than a transparency map, because the model outputs coverage and the correct answer is not coverage.

Unusual subjects. The models are excellent on people, common animals, cars and everyday products, because that is what the training sets contain. Machinery, textiles, food in a bowl, agricultural equipment and anything culturally specific are markedly weaker. If your work is in a domain the training data under-represents, expect to finish by hand more often.

The output resolution ceiling. Many models run internally at 320×320 or 1024×1024 and upscale the mask. On a 45-megapixel file the returned edge is therefore soft and approximate regardless of how sharp the original is. Tools that handle this well run a second high-resolution refinement pass; tools that do not give you a slightly mushy outline that no amount of sharpening fixes.

The workflow that uses them properly

Do not treat the output as the answer. Treat it as ninety seconds saved.

1. Run the model. Get the mask.

2. **Look at the mask alone**, Alt/Option-clicking the thumbnail. Every failure above is obvious in black and white and invisible against the picture.
3. Fix the enclosed gaps with a hard black brush.
4. Fix the thin structures by hand, or accept losing them if they do not matter.
5. Check the edge at 100 per cent for the resolution ceiling; if it is soft, refine it against the original with a channel-based mask or a path along the hard sections.
6. Decontaminate the colour, which is the next lesson.

The five minutes of checking is what separates a listing photograph that looks professional from one that looks automatic, and it is the part that is skipped.

The part that is genuinely unsettled

These models were trained on large image collections scraped from the web. Whether that training was lawful, and whether the models' outputs carry any obligation to the people whose images were used, is being argued in several jurisdictions at once and the answers differ by country. Some datasets were assembled with consent and licensing; many were not, and some have been withdrawn after audits found material that should never have been in them.

There is no settled position to give you here, and anyone who tells you there is has picked a side in an argument that courts have not finished. What is within your control is narrower and worth doing: prefer models with a stated licence and a documented training set, keep client images off third-party servers where confidentiality applies, and do not represent a model-generated result as a measurement of anything.

BEFORE YOU MOVE ON

A one-click background removal on a photograph of a bicycle returns a clean outline, but the spaces between the spokes and inside the frame triangle are solid. What does this indicate about how the tool works?

A. The mask resolution was too low to resolve the spokes, and running the same model at full resolution would recover the gaps.
B. The tool applied a fill-holes post-processing step, which can be disabled in most implementations to keep the gaps open.
C. The model predicts a plausible object silhouette learned from training examples, and most of those examples had solid bodies, so enclosed background is filled rather than detected.
D. Background removal works on colour similarity to the frame edge, so enclosed regions are never reached by the flood from the outside.

27 · 9 min

Three equations, seven unknowns: why hair is hard

THE ONE THING TO KEEP

A boundary pixel is a mixture of foreground and background whose separation is mathematically underdetermined, so every extraction needs extra information, and the edge colour must be recomputed rather than contracted, feathered or contrasted away.

The compositing equation

Every pixel in a photograph where subject meets background obeys one relationship:

```
I = α·F + (1 − α)·B
```

`I` is what the camera recorded. `F` is the foreground's true colour at that pixel, `B` is the background's true colour, and `α` is what fraction of the pixel the foreground covers.

That is three equations — one per colour channel — and seven unknowns: three for `F`, three for `B`, and `α`. The system is underdetermined. There is no correct answer derivable from the pixel alone; there are infinitely many combinations of foreground, background and coverage that produce the same recorded value.

This is not a software limitation. It is why extracting hair is genuinely hard, why every tool needs extra information from you, and why the answer is always an estimate.

How the extra information is supplied

A trimap. You mark three regions: definitely foreground, definitely background, and unknown. The algorithm now has known `F` values nearby and known `B` values nearby, and only has to solve for `α` in the unknown band. This is exactly what the **Refine Edge brush** in Select and Mask is doing when you scrub over the hair — you are drawing the unknown region of a trimap.

A known background. Shoot against a uniform colour and `B` is known everywhere, which collapses the problem. That is chroma key, and it is why studios use it.

A learned prior. Deep matting models estimate `α` from the image using patterns learned from thousands of hand-made mattes. Better than the classical methods on hair and confidently wrong on anything unusual.

Colour spill, and why contracting does not fix it

Now the second half of the problem. The equation says a partial pixel's recorded colour `I` is a mixture that **includes the background**. Extract the alpha correctly and place the subject on white, and those partial pixels still carry the old background's colour in their `F`.

Against a green wall, hair edges are genuinely green. Against a red curtain, they are pink. Against a grey studio wall, they are grey — which is the familiar dull halo.

Three things people try, and why each fails:

- **Contracting the selection** by a pixel. This does not remove contamination; it removes the hair. Fine strands are only two or three pixels wide, so contracting deletes them entirely, and you get the helmet.
- **Feathering.** This blends the contaminated pixels further into the surroundings, spreading the problem over a wider band.
- **Raising the mask contrast.** This turns partial pixels into fully-on or fully-off ones, which removes the softness that made the edge believable.

The only fix that addresses the cause is **colour decontamination**: recomputing the `F` of each partial pixel from the nearby known foreground colour, while keeping its `α`. In Photoshop this is the *Decontaminate Colours* checkbox in Select and Mask, and it outputs to a new layer because it rewrites pixel colours. In video compositing it is called **spill suppression** and every keyer has one.

A green fringe on hair: three things that fail, and what addresses the cause

Does not work	Works
• Contracting by a pixel — strands are two or three pixels wide, so you delete them and get the helmet	• Decontaminate Colours — rewrites each partial pixel's colour from nearby known foreground and keeps its alpha
• Feathering — spreads the contaminated pixels across a wider band	• In GIMP: a layer clipped to the cut-out, set to Colour mode, painted along the fringe at 30 to 50 per cent
• Raising the mask contrast — turns partial pixels fully on or off and removes the softness that made the edge believable	• At capture: a neutral grey wall, subject two metres clear of it, lit separately, so there is little spill to remove

The alpha is correct. The colour underneath it is wrong, because the recorded value of a partial pixel is a mixture that includes the old background. Only recomputing that colour treats the cause; everything on the left treats the symptom and costs you the hair.

The free route: GIMP does not have an automatic decontaminator, and the manual equivalent works. Create a new layer clipped to the cut-out, set it to **Colour** blend mode, sample the subject's correct colour near the edge, and paint along the contaminated fringe at 30 to 50 per cent opacity. You are replacing the hue while keeping the luminosity and the alpha. It takes two minutes on a portrait and it works.

A second manual trick for green spill specifically: a Hue/Saturation layer targeting Greens, saturation down, masked to the edge band. Crude, fast, and adequate for a marketplace listing.

Shooting to make this easy

The cheapest fix happens before the shutter.

- **Shoot against a background close in tone and neutral in colour** to what the subject will end up on. Grey or white for a product going on white. Contamination from a neutral background is far less visible than from a coloured one.
- **Separate the subject from the background** by two metres if you can, and light them separately. This reduces the coloured light bouncing off the background onto the subject, which is spill that no decontamination can fully remove because it is real light that really landed on the hair.
- **Green screen is for video, not stills.** Video needs it because frame-by-frame manual work is impossible. For a single still, a plain grey wall plus ten minutes of masking gives a better result with no spill problem.

Checking the result

The test that catches everything: put the cut-out on three backgrounds in turn — white, mid-grey, and a saturated colour that contrasts with the original background. The saturated one is where contamination screams. If the hair reads as clean on all three, it is clean.

Then look at the edge at 100 per cent against the darkest of them. A pale rim you did not see on white will be obvious on black.

What remains unsolvable

Some pixels have no answer. If the subject is motion-blurred against a background, the mixing happened over time as well as space, and the equation's `F` is itself a blur of several different foreground colours. If the background behind the hair is itself detailed and high contrast — a bookshelf, foliage — then the contaminating colour varies pixel to pixel and decontamination has no consistent value to replace it with.

In those cases the honest answer is that the shot cannot be cut out cleanly, and the choices are to keep some of the original background as a soft vignette, to reshoot, or to tell the client. Spending three hours on a shot that was not extractable is a mistake most people make once.

BEFORE YOU MOVE ON

A subject shot against a green wall is cut out with a technically accurate alpha channel and placed on white. The hair shows a green fringe. The editor contracts the mask by one pixel, and the fringe reduces but the fine strands vanish. What does this demonstrate?

A. Contracting removes the partial pixels entirely, and those pixels are the hair, so the contamination and the detail are the same pixels and cannot be separated by erosion.

B. The alpha channel was inaccurate after all, and a better matting model would produce an edge with no contamination.

C. Green is the channel with the highest luminance weight, so green contamination is uniquely resistant to erosion in a way other colours are not.

D. The mask needed feathering rather than contracting, which would blend the green into the white background and make it invisible.

28 · 9 min

The four subjects that are not opaque

THE ONE THING TO KEEP

Non-opaque subjects split into coverage, transmission and emission, and each needs a different method — matting and decontamination for hair, Multiply and Screen plates for glass, and a plain Screen with no mask at all for smoke and flame.

Different problems that look like one problem

Hair, fur, glass and smoke all resist the ordinary cut-out, and for three different reasons. Naming the reason tells you the method.

- **Hair and fur** are opaque, and *thin* — each strand covers part of a pixel. The problem is fine partial coverage.
- **Glass** is transparent and *refracting* — it shows the background through it, distorted, plus reflections on top.
- **Smoke, steam, flame and water spray** are *additive or subtractive light*, not coverage at all.

Hair, in three passes

The method that works on a real portrait, and it is deliberately not one operation.

Pass 1 — the body. Quick Selection or a segmentation model for the head and shoulders, then Quick Mask and a hard brush to fix the solid outline. Ignore the hair completely.

Pass 2 — the wispy zone. Select and Mask, and scrub the Refine Edge brush *only* along the transition band. Turn on *Decontaminate Colours*. Output to a new layer with a mask. Do not touch the smooth parts of the outline; the refine brush makes clean edges worse.

Pass 3 — the strands that are still missing. Some fly-away hairs will not survive any method. The professional answer is to paint them, with a small hard brush at one to two pixels, sampling the hair's colour, on a layer under the subject, following the direction the real strands went. Five or six strokes. Photoshop and Krita both have brush presets that scatter multiple hair-like strokes at once; Krita's are better and free.

Where the background behind the hair is busy, the alternative is not to cut out at all: keep a soft halo of the original background and blur it into the new one. Against a dark, plain new background this is invisible and takes two minutes.

Fur: the same, with more of it

Fur differs in that there is far more transition zone — an animal is mostly edge. Two things help:

- Build the mask from the channel with the best separation, not from a selection tool. A ginger cat against a green lawn separates beautifully in the red channel.
- Accept a slightly soft outline. Fur is genuinely soft, and a crisp one is what looks wrong.

Glass, and why a mask is the wrong tool

A wine glass, a bottle, a window, a pair of spectacles. The correct answer is almost never a cut-out.

The blend-mode method, which is what product retouchers actually use:

1. Shoot the glass against **white** and again against **black** if you can. Two frames, same position, tripod.
2. The black frame holds the reflections and highlights — everything the glass *adds*. Put it on top and set it to **Screen**, which makes its black areas invisible.
3. The white frame holds the shadows and absorption — everything the glass *removes*. Put it below and set it to **Multiply**, which makes its white areas invisible.

4. The new background goes underneath both. The glass now genuinely transmits it: the multiply layer darkens it where the glass is thick, the screen layer adds the highlights on top.

This is the technique, it is forty years old, and it produces results no mask achieves, because it reproduces what glass physically does to light rather than pretending glass is a shape.

With only one frame, a cut-down version works: mask the glass's silhouette, set the glass layer to a blend mode that lets the new background through, then paint the specular highlights back on a Screen layer above. It is a compromise and it is usually good enough for a listing.

Smoke, steam and flame

These are light sources, not objects. Shot against black, smoke is pure addition: the black background contributes nothing and the smoke contributes everything.

So: put the smoke plate over your scene, set it to **Screen**, and the black vanishes with no mask at all. Adjust with a Levels layer on the smoke to control how much shows. For dark smoke against a bright sky, shoot it against white and use **Multiply** instead.

This is why every stock library sells smoke, fire, dust, rain, light leaks and bokeh overlays photographed against black. They are designed to be screened, and a mask would destroy exactly the soft variable density that makes them look real.

Free plates: Pexels and Pixabay carry usable smoke and bokeh footage stills under permissive licences. Check the specific licence for commercial use rather than assuming.

Water, net, lace and mesh

A hybrid case. Netting, lace, mosquito mesh and chain-link are geometrically regular and genuinely partially transparent. The route that works is to build a mask from the channel with the strongest contrast, push it hard with Levels to get the pattern clean, and then *reduce its opacity* to reinstate the partial transparency rather than trying to capture the partial values directly. The regularity that makes them hard to select is also what makes a pushed mask acceptable.

The honest summary

There is no single method for non-opaque subjects, and the tools that claim there is are selling the easy eighty per cent.

The decision, each time:

- Can I shoot this again, better? Almost always the cheapest fix.
- Is this coverage (hair, fur) — then matting and decontamination.
- Is this transmission (glass, water) — then blend modes and, ideally, two frames.
- Is this emission (smoke, flame, flare) — then Screen, with no mask.
- Is none of it working — then say so before the deadline rather than after.

And the limitation nobody can engineer round: a single photograph of a transparent object against a complex background does not contain enough information to separate the object from what is behind it. The information was destroyed at capture, when three numbers per pixel were recorded for something that needed six. Every technique in this lesson is a way of supplying the missing information from somewhere else — a second frame, a known background, a physical model, or your hand.

BEFORE YOU MOVE ON

A photographer has a shot of wood smoke against a black background and needs it over a daylight street scene. Which approach reproduces the smoke correctly, and why?

A. Build a soft mask from the smoke's luminosity and apply it to the smoke layer in Normal mode, so the density of the mask matches the density of the smoke.
B. Place the smoke layer over the scene in Screen mode, because Screen makes black contribute nothing and the smoke adds light exactly as it did in the original frame.
C. Run a segmentation model on the smoke to produce an accurate alpha channel, then composite in Normal mode.
D. Place the smoke over the scene in Multiply mode, so the darker parts of the frame show through and the smoke sits behind the highlights.

29 · 8 min

Four hundred cut-outs, and the twelve that are wrong

THE ONE THING TO KEEP

A batch of cut-outs is a pipeline plus a sampling check — a grey-background contact sheet of the masks finds the errors a white-background grid hides — and the predictable failure categories should be pulled out for manual work before the batch runs, not after.

The job that actually arrives

Nobody pays for one cut-out. The work is a shop with 400 products, a catalogue reshoot, a marketplace migration where every listing needs a white background at 2000 pixels square.

Doing 400 by hand at four minutes each is 27 hours. Doing them automatically and checking properly is about four. The difference is entirely in how you organise the checking, because the automatic pass will get somewhere between 85 and 97 per cent of them right and you cannot tell which without looking.

The pipeline

```
# 1. one folder in, one folder out, consistent model
rembg p -m isnet-general-use ./raw ./cutouts

# 2. composite onto white at a fixed canvas, keeping proportion
magick mogrify -path ./final \
  -background white -alpha remove -alpha off \
  -resize 1800x1800 -gravity center -extent 2000x2000 \
  -quality 88 -strip ./cutouts/*.png
```

That is the whole thing, and both tools are free. `rembg` does the masks; ImageMagick flattens onto white, resizes to fit inside 1800 pixels, centres on a 2000-pixel canvas with a consistent margin, and strips metadata.

The `-resize 1800x1800` before `-extent 2000x2000` is what gives every product the same 100-pixel breathing space regardless of its shape. Without it, a tall bottle and a wide box end up at different visual scales and the grid looks wrong even though every file is the same size.

If you prefer a graphical route: GIMP's *Script-Fu* console and BIMP plugin both batch, Photoshop has Actions plus *File* > *Automate* > *Batch*, and Photopea has a scripting console. The principle is identical.

Consistency is the deliverable

For a catalogue, the individual cut-out matters less than the set matching. Decide these once and encode them in the script, not in your judgement:

- **Canvas size and aspect.** Most marketplaces want square; several require a minimum of 1600 pixels; some require the product to occupy 80 to 85 per cent of the frame.

- **Margin.** Same on every file.
- **Background.** Pure 255 white if the platform demands it; around 250 grey on your own site, so white product highlights do not dissolve into the page.
- **Shadow.** Either every product has one or none does. A grid where half the items float is the commonest tell of a job done in two sittings.
- **File naming.** `sku_view_index.jpg`, machine-readable, no spaces.

Checking without looking at 400 images

Three passes, in increasing cost.

Contact sheet. Build a grid of every output and look at it at once.

```
magick montage ./final/*.jpg -tile 10x -geometry 200x200+4+4
sheet.jpg
```

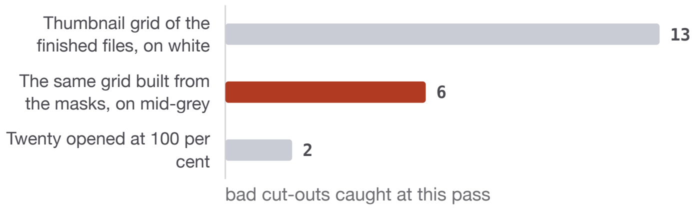
At thumbnail size you will catch the catastrophic failures instantly: a product that came out as a silhouette, one that lost its handle, one where the model grabbed the shadow instead of the object, one that is upside down. This finds perhaps two thirds of the errors in three minutes.

Alpha contact sheet. The same grid built from the masks on a mid-grey background rather than white. Errors that vanish against white — a filled-in handle gap, a chewed edge — are obvious against grey. This is the pass most people skip and it is the one that pays.

Full-size sample. Open twenty at 100 per cent, chosen to cover the variety: the shiniest, the darkest, the thinnest, the one with a transparent lid, the one shot on a different day. If all twenty pass, the batch is probably sound. If three fail, inspect the whole set.

That is sampling, and it is the correct use of it: a sample tells you whether there is a systematic problem, not whether every individual file is right.

Where the twenty-one bad cut-outs in a batch of 400 were found



The grey mask sheet is the pass most people skip and the one that pays: a filled-in handle gap and a chewed edge both vanish against white and shout against grey. Twenty opened at full size is a sample, and a sample tells you whether there is a systematic problem, not whether every file is right.

The categories that will fail, predictably

Pull these out for manual work *before* running the batch, and the batch gets much better:

- **Glass, bottles, anything transparent.** Blend-mode work from the previous lesson.
- **White or very pale products on white.** The model has no contrast to work with.
- **Products with holes** — handles, rings, open boxes, mesh.
- **Anything with fine structure** — jewellery chains, lace, wire.
- **Reflective metal**, where the reflection of the studio is part of the product.

Sorting the shoot into "automatic" and "by hand" folders at the start turns a frustrating batch into two predictable ones.

Privacy, and what leaves your machine

Running `rembg` locally means nothing is uploaded. Using a web service means 400 of a client's product photographs, possibly unreleased, go to a third party under terms you probably have not read, and several such services reserve the right to use uploads for improvement.

For public catalogue images this is a small risk. For a product not yet announced, for anything under a non-disclosure agreement, or for photographs containing people, the local route is not merely cheaper — it is the only one you can honestly tell a client you used.

The limitation that determines the price

An automatic batch does not produce professional cut-outs. It produces acceptable ones, quickly, most of the time.

The gap is in the last five per cent: the sub-pixel edge, the decontaminated fringe, the contact shadow that makes the object sit on the surface. For a marketplace grid viewed at 200 pixels, nobody sees the gap. For a printed catalogue, a hero image or a brand that cares, they do.

Knowing which job is which is how you price the work. Quoting a by-hand rate for a job that a batch would have satisfied loses you the job; quoting a batch rate for a job that needed the hand work loses you the weekend.

BEFORE YOU MOVE ON

After batch-removing backgrounds from 400 product photographs, an editor builds a contact sheet of the finished images on white and sees nothing obviously wrong. Why is that insufficient evidence that the batch succeeded?

A. Contact sheets compress the images, and compression artefacts mask edge errors that are present in the full-size files.

B. Errors that leave white where the background was — filled gaps, chewed edges, lost thin parts — are indistinguishable from a correct result against a white sheet.

C. A sample of 400 is too small to be conclusive, and every file has to be opened individually before delivery.

D. Contact sheets show the composite rather than the alpha channel, so they cannot reveal whether the files were saved with transparency at all.

CASE STUDY · MODULE 3

Four hundred and twelve brass items, and the twelve that were not

A two-person brassware export business in Moradabad, migrating its whole catalogue to a marketplace that requires pure white backgrounds.

The job that actually arrived

Four hundred and twelve items, photographed over three years by three different people against three different grey walls. The new marketplace wanted 2,000-pixel squares on white, the product occupying most of the frame, one main image per listing. Nobody had ever cut any of them out.

A freelancer in Delhi quoted per image, by hand, five working days. The quote was defensible and the money was not in the budget for a migration that was itself a cost. The alternative was the free pipeline: `rembg` for the masks, ImageMagick to flatten onto white and pad every item to the same canvas so a tall lamp and a wide tray sit at the same visual scale in the grid.

There were two complications, and they pulled in opposite directions.

Ninety of the items are polished brass. A polished surface is an image of whatever is in front of it, which in these photographs was a window, a tripod and a man in a blue shirt. A segmentation model outputs coverage, and coverage is not the right answer for a surface whose appearance is mostly reflection.

Eighteen items were an unreleased range under an agreement with a buyer in Hamburg. Running them through a web background remover means uploading photographs of an unannounced product line to somebody else's server, under terms of service nobody in the business had read, several of which reserve the right to use uploads for improvement.

Three routes

A web service. Fastest by a distance, and the free tier returns a small image while the full-resolution file is paid. It also disposes of the confidentiality question by answering it badly.

`rembg` **locally.** Two hours to install on a machine with no graphics card, then two to five seconds per image on the CPU, unlimited, nothing uploaded. It would still fail on the reflective items, on the teapots with handle gaps, and on the pierced lattice lamps, because those are the predictable failures of a salient-object model.

Pay the freelancer for all of it. Five days, consistent quality, a bill, and no learning that stays in the business.

The decision that actually mattered was not which tool. It was whether to sort the catalogue into *automatic* and *by hand* folders **before** running the batch, which costs a morning of looking at 412 thumbnails and deciding, or to run everything and sort out the mess afterwards.

The part that is not about cut-outs at all

For a catalogue, the individual cut-out matters less than the set matching, and that is a set of decisions to be made once and written into the script rather than exercised four hundred times.

The canvas had to be 2,000 pixels square, because the marketplace said so. The product had to be resized to fit inside 1,800 first and then centred on that canvas, which is what gives a tall lamp and a wide tray the same hundred pixels of breathing space and stops the grid looking wrong even when every file is technically the same size. The background had to be exactly 255 white, because the platform checks automatically and rejects what is not. Shadows had to be present on everything or on nothing — a grid where half the items float is the clearest possible sign of a job done in two sittings by two people. And the file names had to be machine-readable, because 412 of them were going into a spreadsheet.

None of that is a skill. All of it is the difference between a set that looks commissioned and one that looks assembled, and it takes twenty minutes to decide and two seconds to encode.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

They spent the morning sorting. Ninety reflective items, eleven lattice lamps, seven trays with pierced rims and four glass-lidded jars went into a *by hand* folder. The remaining 300 went through `rembg` and an ImageMagick line that resized to fit 1,800 pixels and centred that on a 2,000-pixel white canvas, so every product got the same margin regardless of shape.

The checking was three passes. A contact sheet of the finished files at 200 pixels found thirteen catastrophes in about three minutes: two items that came out as silhouettes, one where the model had taken the shadow instead of the object, one upside down. A second contact sheet built from the masks on a mid-grey background found six more that the white sheet had hidden — two teapots whose handle gaps had been filled in, and a spice box whose edge had been chewed. Twenty opened at 100 per cent found two with the soft, approximate outline that comes from a model running at its own internal resolution and scaling the mask up.

Twenty-one bad cut-outs out of 300, all found, in under half an hour of checking. The reflective items were done by hand over the following two days using a Multiply plate and a Screen plate rather than a mask, which is the only thing that works on brass.

The marketplace then rejected four files on a rule nobody had read: the product had to occupy at least eighty per cent of the frame, and four thin items did not. The pipeline was changed in one place — the resize target — and rerun for those four.

The sentence they could put in an email to Hamburg, truthfully, was that no photograph of the new range had left the office.

The grey contact sheet built from the masks found six errors the white one had hidden. Why?

Both sheets show the same files, but against white the two commonest mask faults are invisible. A filled-in handle gap is white where it should be transparent, and on a white background white and transparent look identical. A chewed edge eats a few pixels of product and replaces them with white, which against white is also nothing to see. Composited on mid-grey, both jump out immediately because the transparent regions are now a different tone from the page. It is the pass most people skip, and it is the one that finds the failures characteristic of the model rather than the failures characteristic of the photograph.

Which categories should be pulled out before a batch runs, and what is the mechanism behind each failure?

Transparent and reflective objects, because the model outputs a per-pixel coverage value and the correct answer for glass or polished brass is not coverage at all — it is transmission and reflection, which need Multiply and Screen plates rather than a mask. Products with enclosed gaps — handles, rings, open boxes, mesh — because a model trained on human-drawn masks of solid bodies tends to produce closed, plausible silhouettes. Anything with fine structure such as chains, lace or wire, because those few pixels contribute almost nothing to the training objective, so the model is barely penalised for losing them. And very pale products on a pale background, where there is no contrast to work from.

What did running the models locally buy the business, beyond the money?

Two things. First, a statement they can make to a customer and stand behind: no photograph of the unreleased range left the premises, which is not a claim you can make about a web service whose terms you have not read and several of which permit training on uploads. Second, independence: the pipeline is two commands on their own machine, with no per-image cost, no queue, no price change and no service that can be withdrawn. The cost was two hours of installation and a slower per-image time on a machine with no graphics card, which for 300 images at a few seconds each was irrelevant.

MODULE 3 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A pixel at the edge of hair against a wall is genuinely about 40 per cent hair. Why can no tool derive the correct foreground colour from that pixel alone?

A. The pixel was compressed by JPEG and its value is unreliable
B. Three recorded numbers cannot fix seven unknowns
C. Alpha is stored at half resolution in most file formats
D. Anti-aliasing has already averaged four neighbouring pixels together

2 A subject cut from a green wall shows a green rim against white. Contracting the selection by a pixel removes the rim and most of the fine hair with it. What addresses the cause?

A. Applying a one-pixel Gaussian blur to the whole cut-out layer before export
B. Feathering the mask by two pixels so the rim is less abrupt
C. Raising the mask's contrast so no pixel is partly selected
D. Rewriting each partial pixel's colour, keeping its alpha

3 You need a selection of sky seen through bare branches. Why is Colour Range a better tool here than the Magic Wand?

A. It returns partial values, so the twigs' edge pixels are partly selected
B. It selects only pixels connected to the one you clicked, so the branches are excluded
C. Its tolerance slider reaches higher numbers than the Magic Wand's
D. It runs a segmentation model rather than comparing colour values

4 You lift the shadows in a landscape through an inverted luminosity mask and get no halo along the roofline, where a hard selection always produced one. What prevented it?

A. The software feathered the mask to about two pixels
B. Luminosity masks apply in linear light rather than on encoded values
C. The mask varies continuously, so there is no boundary
D. Shadows are inherently less prone to halos than highlights are

5 An automatic cut-out of a teapot comes back with the gap between the handle and the body filled in. Which property of the model explains it?

A. It runs at a reduced internal resolution and upscales the mask afterwards
B. It predicts plausible silhouettes, and most were solid
C. It ran out of colour tolerance and treated the gap as subject
D. It chose the larger of two objects because it can only return one

6 You check 400 automatic cut-outs by opening twenty at full size, chosen to cover the hardest cases, and all twenty pass. What have you established?

A. That the remaining 380 are almost certainly fine
B. That the cheaper contact sheet passes can be skipped on this job
C. That the model's confidence threshold was set correctly
D. That there is no systematic problem in the batch

MODULE 4

Tone and colour

Light and colour are where a photograph is actually made or lost, and where most of the reversible damage happens. This block covers reading the measurement rather than trusting the screen, white balance when there are two light sources, shaping form by hand with dodge and burn, what local contrast tools are really computing, saturation that does not clip, monochrome conversion that separates rather than flattens, grading with curves and lookup tables, skin tones without a false universal standard, and matching forty photographs so a catalogue reads as one shoot.

> **BY THE END OF THIS MODULE YOU CAN**
>
> Take a set of unevenly lit photographs to a consistent, defensible tonal and colour result — diagnosing casts from per-channel evidence, correcting mixed lighting locally, shaping form without banding or halos, and stating what you measured rather than what looked right on your screen

30 · 10 min

Brightness/Contrast is one knob for a problem with five

THE ONE THING TO KEEP

Set the black and white points first, shape the middle with Curves, and remember that contrast in the midtones is borrowed from the shadows and highlights.

Read the histogram before you touch anything

The histogram is a bar chart of your image: how many pixels sit at each brightness from black on the left to white on the right. It is the only honest report you have, because your screen brightness, the room you are sitting in and your own eyes all lie.

Three things to read off it in two seconds:

- **Data piled against the left wall** — shadows are clipped. Those pixels are pure black and hold nothing.
- **Data piled against the right wall** — highlights are blown. Same problem at the other end, and far more visible, because clipped highlights read as a hole in the picture.
- **A gap at one end** — the image never reaches black or never reaches white. That is the classic flat, hazy, grey photo, and it is the easiest fix in this course.

What Brightness/Contrast does, and why it is not enough

Brightness/Contrast gives you two controls over the whole image at once. Raise brightness and every tone moves up together, including the ones already close to white, which run out of room and clip. Raise contrast and it pushes the ends apart from the middle.

In fairness, Photoshop's version stopped being a crude linear shift in CS3, around 2007, and now protects the extremes somewhat. But the shape of the problem does not change. A dull market photo usually needs the shadows deepened *and* the midtones lifted *and* the highlights left alone. That is three separate moves. One knob cannot do it, so what happens is that you push brightness until the faces look right and discover the sky has turned into white paper.

Levels and Curves are one tool at two levels of control

Levels: three numbers

Levels (Ctrl/Cmd+L, or better, a Levels adjustment layer) gives you the histogram with three sliders under it.

- **Black point** (left): drag it right until it sits where the data actually begins. Everything to its left becomes pure black. This one move fixes most flat images.
- **White point** (right): drag it left to where the data ends.
- **Midtone / gamma** (middle): drag left to lighten the middles, right to darken them, without moving the two ends you just set.

Hold **Alt/Option** while dragging the black or white slider in Photoshop and the preview shows only the pixels you are about to destroy. Stop when the first specks appear. Photopea has this too, and GIMP's Levels dialogue has the same three sliders.

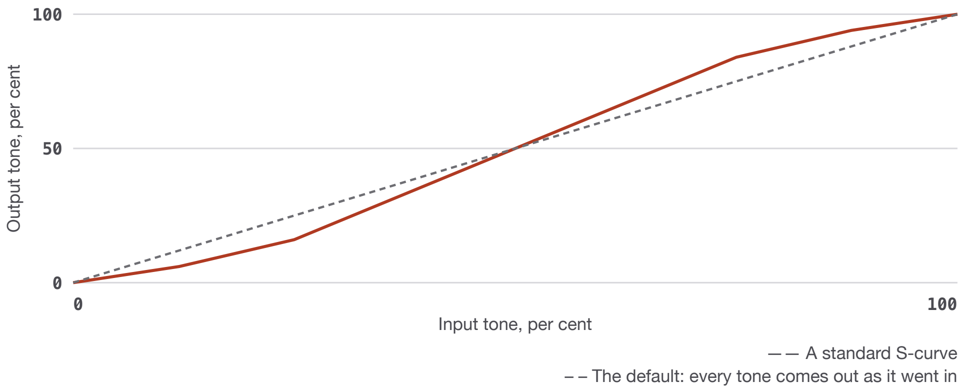
Curves: the same idea with real control

Curves is Levels plus the ability to bend the middle. The horizontal axis is the input tone, the vertical is the output. The default is a straight diagonal: every tone comes out as it went in.

- Click the middle of the line and pull it down half a centimetre. The whole image darkens, ends held.
- Click a quarter of the way from the bottom, pull down; click three-quarters up, pull up. That is the **S-curve** — the standard contrast move, and it is what "punchy" means.

Understand what the S-curve costs. Contrast is not created, it is borrowed. Steepening the middle of the curve spreads the midtones over more output levels, and the only place that room can come from is the ends, which get flattened. That is why heavy contrast eats detail in shadows and highlights. Look at the top and bottom of your curve, not just the middle.

What an S-curve spends, and where it spends it



Between a quarter and three quarters of the input, the curve is steeper than the diagonal: half the input range now covers two thirds of the output, which is what punchy means. That room is borrowed from the ends, which flatten — the darkest twelve per cent of the input is squeezed into six per cent of the output. Look at the top and bottom of your curve, not only the middle.

There is also a **targeted adjustment tool** — the hand icon in the Curves panel. Click it and drag directly on the photograph, and it moves the curve point for that tone. The fastest way to darken *that* wall without knowing where the wall sits on the curve.

Colour casts live in the channels

The dropdown at the top of the Curves panel switches from RGB to Red, Green or Blue individually. This is where colour correction actually happens.

The relationships are worth memorising, because they are just opposites: less blue is more yellow, less green is more magenta, less red is more cyan.

So a photo shot under a tube light with a green cast: switch to the Green channel, pull the middle of the curve down slightly. A sunset that has gone cold: lift the Red curve's midpoint and drop the Blue's. Small moves — a few pixels of drag is a lot.

Faster still, when there is something in the frame you know to be neutral: use the grey eyedropper in Levels or Curves and click a grey wall, a white shirt in shade, a sheet of paper. The software calculates what correction makes that pixel neutral and applies it to the whole image. Choose a mid-grey, not a bright white, which is often clipped and therefore useless as a reference.

If you shot raw, do this in the raw developer instead — **darktable** and **RawTherapee** are free and their white-balance control is working with the original sensor data before it has been squeezed into 8 bits. That is strictly better than fixing it later.

The 8-bit tax

Every stretch you apply to an 8-bit image spreads 256 levels over a wider range, leaving gaps. Open the histogram after a strong Levels move and you will see it has turned into a comb — white vertical lines where no pixels remain. In a photograph of a market you will never notice. In a smooth gradient — a sky, a studio backdrop, a wall — those gaps show up as visible bands.

Two defences. Convert to 16-bit before big tonal work (**Image > Mode > 16 Bits/Channel**), which gives you 65,536 levels per channel and enough room to stretch. Or start from a raw file, which was 12 or 14 bits at capture.

Today: open your flattest photo, add a Levels adjustment layer, and move only the black-point slider to where the histogram begins. Nothing else. That one move beats an hour of sliders.

BEFORE YOU MOVE ON

A market photo looks flat and grey. Someone pushes Brightness and Contrast up until it looks punchy on screen. The sky loses all its cloud detail and the shadows still look muddy. What is the mechanism?

A. The sky was overexposed in camera, so the cloud detail was never in the file and no adjustment could have kept it.
B. One control moves the whole tone range together, so by the time the midtones look right the highlights have been pushed past white and cannot come back, while the shadows were never given their own move.
C. The image needs more saturation rather than more contrast; the flatness is a colour problem, not a tonal one.
D. Brightness/Contrast should have been applied on a duplicated layer, which would have prevented the loss.

31 · 9 min

There is no correct histogram

THE ONE THING TO KEEP

The histogram only reports value distribution, so its real jobs are catching per-channel clipping and missing tones — and the camera's version is computed from the JPEG preview, which is why the raw file usually holds highlights the warning says are gone.

What the graph is, precisely

The histogram counts pixels by value. The horizontal axis runs from 0 to 255; the height at each position is how many pixels hold that value. That is all. It has no idea where in the frame those pixels are, what they are pictures of, or whether the photograph is any good.

The first thing to unlearn is the idea of a correct shape. A photograph of a black cat in a coal cellar should be piled on the left. A high-key portrait on white should be piled on the right. A foggy morning is a narrow lump in the middle and that is what fog looks like. An editor who drags every image towards a even spread is destroying the subject.

What the histogram genuinely tells you is whether data has been **lost** at the ends, and whether tones are **missing** in between.

RGB, colour, and luminosity are three different graphs

Most editors offer several histogram views and the difference matters.

- **Luminosity** — one graph, weighted 0.2126 / 0.7152 / 0.0722, showing perceived brightness. The right one for judging exposure.
- **Colours** or **RGB overlay** — the three channels drawn on top of each other. The right one for spotting a cast and, crucially, for spotting *per-channel* clipping.
- **RGB** in Photoshop's menu is a combined graph that is not the same as luminosity and is the least informative of the three.

Per-channel clipping is the one that catches people. A photograph of a deep red flower can look perfectly exposed on the luminosity histogram while the red channel is jammed against 255 and every petal has lost its internal detail. Nothing on the luminosity graph reveals that. Switch to the three-channel view and it is unmissable.

This is why heavy saturation is dangerous: raising saturation pushes channels apart, and the leading one hits the wall first. Check the colour histogram after any saturation move.

The clipping warnings, and what they mean

Most editors will show clipped pixels in a flashing colour. In darktable and RawTherapee they are on by default; in Lightroom the triangles at the corners of the histogram toggle them; in Photoshop, hold Alt/Option while dragging a Levels endpoint and the preview shows only what you are about to destroy.

Read them properly:

- **Specular highlights** — the sun's reflection on chrome, a bare bulb — are *supposed* to be pure white. There is no detail in them and never was. A clipping warning there is not a problem.
- **A blown sky, a blown white shirt, blown skin** are real losses. Once all three channels are at 255 there is nothing to recover.
- **Clipped shadows** are often fine. A few per cent of pure black in a night scene gives the picture somewhere to rest. A clipped shadow across half the frame has eaten the subject.

The histogram your camera showed you is a lie

A fact that changes how people shoot. The histogram on the back of a camera, and the blinking highlight warning, are computed from the **embedded JPEG preview**, not from the raw data. That preview has the camera's contrast curve, saturation and white balance applied.

So the camera can warn you that the sky is blown when the raw file still holds a stop of it. In practice the raw file typically has between half a stop and a stop and a third more highlight headroom than the camera's display suggests, depending on the model and the picture style.

The practical consequence is the technique called **expose to the right**: deliberately push exposure up until the camera warns, because the raw file will still hold it, and a brighter raw file carries more signal relative to noise in the shadows. The risk is obvious and the discipline is to test your own camera once — bracket a scene, develop the raws, find where the recovery actually stops — rather than trusting a general rule.

Setting a flat, low-contrast picture style in the camera makes the preview histogram closer to the raw and costs nothing, because the raw is unaffected by it.

The waveform, and why video people use it

A histogram discards position. A **waveform** keeps the horizontal position: it plots brightness vertically against the frame's horizontal axis, so you can see that the left side of the shot is clipping while the right side is fine.

For a photograph with a bright window on one side and a dark interior on the other, this is far more useful than a histogram. darktable has a waveform and an RGB parade in its histogram module — click through the modes. Krita has one. Photoshop does not, which is a genuine gap, and the workaround is to use the eyedropper with the Info panel on specific points.

Numbers beat the graph for specific questions

The eyedropper with the Info panel open gives you exact values at a point, and for several questions that is what you want:

- **Is this white shirt actually white?** 240 to 250 in all three channels means printed detail survives. 255 means the texture is gone.
- **Is this grey wall neutral?** Equal values across R, G and B. A difference of five is a visible cast.
- **Are these two product shots matched?** Sample the same point on both.

Set the eyedropper's sample size to **3 by 3 average** rather than Point Sample. A single pixel in a noisy image tells you about the noise rather than the tone.

What the histogram cannot tell you

It cannot tell you where the tones are, which is why two completely different photographs can have identical histograms. It cannot tell you whether the contrast is in the right place — a punchy histogram with all the contrast in the background and none on the face is a bad photograph with a good graph.

And it cannot tell you whether the image is *pleasant*. Every histogram-based rule ever written — fill the range, avoid clipping, aim for a bell curve — produces competent, flat, characterless pictures when followed mechanically. The graph is a measuring instrument that catches specific failures. The decision is still yours, made by looking at the photograph.

BEFORE YOU MOVE ON

A photograph of a red hibiscus looks correctly exposed on the luminosity histogram, with nothing touching either end, yet the petals print as flat shapes with no visible texture. What should the editor check?

A. The image's bit depth, since 8-bit cannot hold enough red values to show petal texture.
B. The three-channel histogram, because the red channel can be clipped at 255 while the weighted luminosity graph still shows headroom.
C. The colour profile, since a wide-gamut file displayed as sRGB flattens saturated reds.
D. The clipping warning, which would have flagged the problem had it been switched on, meaning nothing further needs checking.

---

32 · 9 min

Two light sources, one global slider

THE ONE THING TO KEEP

White balance has a blue-yellow axis and a green-magenta axis, so a green office cast needs tint rather than temperature — and two light sources in one frame cannot be corrected globally at all, only by masking a second correction onto the minority light.

Two axes, not one

White balance has two controls because light varies in two directions.

Temperature runs blue to yellow, measured in Kelvin, and the scale is counter-intuitive: a *low* number like 2700 K is warm orange candlelight, a *high* number like 7000 K is cool blue shade. The number describes the light the picture was shot under, so setting the slider to 2700 K tells the software "this was lit by something orange" and it compensates by adding blue.

Tint runs green to magenta, and it exists because real light sources are not on the blackbody curve. Fluorescent tubes and cheap LED panels are green; some discharge lamps are magenta. A photograph can have a perfect temperature and still be sickly green, and no amount of the temperature slider fixes it, because green is at right angles to the blue-yellow axis.

Most people only ever touch temperature, and the green cast under office lighting is a tint problem. Learning to reach for the second slider fixes half the difficult casts you will meet.

Two casts, two axes, and the slider most people reach for

	Temperature	Tint
bulb, orange cast	**Fixes it** the blue to yellow axis	**No use here** this cast does not lie on this axis
t tube, green cast	**No use here** the cast does not lie on this	**Fixes it** the green to magenta axis

Temperature and tint are at right angles, which is why the green cast under office tubes never yields to the blue-yellow slider. Two light sources in one frame is a different problem again: one global control cannot correct two casts, so set it for the dominant light and mask a second correction onto the other.

The reference that makes it objective

Put something known-neutral in the frame. A grey card costs a few pounds; the white side of a colour checker works; a sheet of plain white printer paper is not neutral but is close enough for most jobs; the back of an unprinted receipt is not.

Shoot one frame of the card at the start of each lighting setup. In the developer, click the white-balance eyedropper on it, then copy that setting to every other frame from the same setup. The whole shoot is now neutral in ten seconds.

Two details that matter. **Sample mid-grey, not bright white** — a white highlight is often clipped, and a clipped channel gives a wrong answer with no warning. And **set the sample size to a 3×3 or 5×5 average**, because one pixel of a noisy card tells you about the noise.

Without a card, look for something in the frame you know is neutral: a concrete wall, a white shirt in shade, newsprint, a stainless steel surface. Sample it and check whether the result looks plausible. A car's silver paint and a whitewashed wall are usually close enough.

Mixed light: why one slider cannot win

The common hard case. A room with a window on one side and tungsten bulbs overhead. The subject's left is lit at 6500 K, their right at 2900 K.

Set the white balance for the window and the tungsten side goes deep orange. Set it for the tungsten and the window side goes deep blue. Split the difference and both sides are wrong, which is what most photographs of Indian weddings, restaurant interiors and hotel lobbies look like.

A global control cannot fix two different casts because there is only one of it. The fix is local:

1. Set the white balance for the **dominant** light — whichever covers more of the subject, usually the window.
2. Add a **Photo Filter** or **Colour Balance** adjustment layer set to cool the remaining orange, or a Curves layer with the blue channel lifted in the midtones.
3. **Mask it** to the tungsten-lit region, with a large soft brush. The boundary between the two lights in the real scene is gradual, so the mask should be too.
4. Check skin across the transition. Skin is where a mismatch shows.

This takes five minutes and it is the single most valuable local correction in interior work.

The better fix is at capture, and it is cheap: **gels**. A sheet of CTO (colour temperature orange) over a flash makes the flash match tungsten; CTB does the reverse. A full sheet costs a few pounds and lasts years. Or simply switch off the tungsten lamps and use the window.

Auto white balance, and why it fails predictably

Automatic white balance works by assuming the scene averages to neutral — the "grey world" assumption — or by looking for the brightest pixel and calling it white.

Both assumptions break on the same kinds of scene, and knowing which tells you when to override:

- **A frame dominated by one colour.** A field of yellow mustard, a red brick wall, a green cricket pitch. The algorithm sees an average that is not neutral, concludes the light was coloured, and removes the colour of the subject. Your yellow field comes out grey.
- **A frame with no neutral in it at all.** Nothing to anchor to.
- **Sunsets.** The whole point of the photograph is that the light is orange, and auto white balance dutifully removes it.

That last one is the useful general principle: white balance is not always meant to be corrected. Firelight should be orange. A blue hour photograph should be blue. "Correct" means "as if under neutral light", which is a technical statement and not always the right one. Set it neutrally first so you know where neutral is, then move deliberately away from it.

Raw, JPEG, and the cost of getting it wrong

In a raw file, white balance is a pair of multipliers not yet applied. Change it and nothing is lost.

In a JPEG it has been baked in, and correcting it means stretching one channel relative to the others inside 256 levels. Correcting a heavy tungsten cast in a JPEG means multiplying the blue channel by something like 2.5, and a blue channel stretched that far bands, goes noisy, and in the darkest areas has nothing left to stretch. The picture can be made neutral, and it will look worse than one developed from raw.

This is the clearest single argument for shooting raw in any situation with unusual lighting — which is most indoor work, most evening work, and every venue with coloured LEDs.

Free tools: darktable and RawTherapee both have temperature and tint with a spot eyedropper and camera presets. GIMP has *Colours > Colour Temperature*, which takes an original and an intended temperature, plus *Colours > Levels* with the grey eyedropper. Photopea has a grey point in Curves and Levels. On a phone, Lightroom mobile's free tier and Snapseed both expose temperature and tint.

One caution on all of them: correcting white balance changes the apparent exposure, because you are scaling channels. Set white balance first, then exposure, then the tone curve. Doing it in the other order means redoing the exposure.

BEFORE YOU MOVE ON

A photograph of a bright yellow mustard field comes out of the camera looking grey and lifeless, although the exposure is correct. What is the most likely cause?

A. The tint slider was set towards magenta, cancelling the yellow of the crop.
B. The camera's automatic white balance assumed the scene averaged to neutral, read the field's yellow as a colour cast in the light, and removed it.
C. The yellow was outside the camera's colour gamut and clipped to a neutral value on conversion.
D. The saturation setting in the camera's picture style was too low for a scene with one dominant colour.

33 · 9 min

Shaping light by hand, on a grey layer

THE ONE THING TO KEEP

Dodge and burn on a 50 per cent grey Soft Light layer at 5 to 10 per cent brush opacity is fully reversible local tonal control, and a steep-curve black-and-white viewing layer is how you see the blotches before your client does.

A darkroom technique that survived

Under an enlarger, you held a piece of card on a wire over part of the print to give it less light — **dodging**, making it lighter — or cupped your hands to give one area more — **burning**, making it darker. Every great print of the twentieth century was made this way. Ansel Adams's negatives were ordinary; his prints were maps of where he had held his hands.

The digital version is the same idea and it is still the most powerful local tool in editing, because it works on *form* rather than on *regions*. A curve applied to a selection has an inside and an outside. Dodging and burning has neither; it has a gradient of emphasis that follows the shape of what is in the picture.

Do not use the Dodge and Burn tools

The tools with those names in the toolbar paint directly on the pixels. They are destructive, they shift saturation as a side effect, and they are hard to modulate. Nobody who does this for a living uses them.

The technique everyone actually uses:

1. **New layer**, filled with **50 per cent grey** (Edit > Fill > 50% Grey).
2. Set its blend mode to **Soft Light** or **Overlay**.
3. Mid-grey is the neutral value in both modes, so the layer is currently invisible.

4. Paint with a soft brush at **5 to 10 per cent opacity**: white to lighten, black to darken, `X` to swap between them.
5. Build up in many passes. Ten strokes at 5 per cent is controllable; one at 50 per cent is a decision.

Everything about that is reversible. Paint the opposite colour to undo locally. Drop the layer opacity to reduce the whole effect. Delete the layer to start again. Add a mask to the grey layer if you want to fence the work.

Soft Light is gentler and better for skin and subtle form. **Overlay** is roughly twice the strength, better for shaping a landscape or pushing a background down.

The same technique works in GIMP (fill with 50 per cent grey, set the layer mode to Soft light), Krita, Photopea and Affinity. No tool is missing here.

The alternative: two curves and two masks

The other standard method, preferred by some retouchers because it is fully re-editable:

- A **Curves layer pulled up**, mask inverted to black — this is your dodge layer. Paint white on its mask where you want it lighter.
- A **Curves layer pulled down**, mask inverted to black — your burn layer. Paint white where you want it darker.

The advantage over the grey layer is that you can change the strength of all your dodging at once by re-editing the curve, long after the painting is done. The disadvantage is two layers and two masks instead of one. Both are correct; pick one and get fluent.

What you are actually doing

Dodging and burning is not brightness correction. It is three separate jobs, and knowing which one you are on stops the work looking arbitrary.

Directing the eye. The viewer looks at the brightest, highest-contrast thing in the frame. Darken the corners and the edges slightly, lift the subject slightly, and the picture reads in the order you intended. This is why almost every advertisement and film poster is darker at the edges.

Building form. A face, a fruit, a fabric fold is a three-dimensional object described only by light. Deepening the shadow side and lifting the lit side makes it rounder. Reversing it makes it flat. This is drawing, and it responds to the same understanding a painter uses.

Evening out what the light did badly. A hot spot from a flash, a bright patch of floor, a distracting white sign in the background. This is correction, and it is the least interesting of the three.

Checking your own work

Dodging and burning at 5 per cent is subtle by design, which means it is easy to produce a blotchy, uneven mess without noticing until the print.

The **viewing layer** is how retouchers check. At the top of the stack, add:

- A **Black and White** adjustment layer, which removes the colour so you are judging tone only.
- Above it, a **Curves layer with a very steep S** — nearly vertical through the middle. This exaggerates every tonal variation enormously.

Switch that group on and your gentle dodging becomes glaringly visible, along with every lumpy edge and every patch you missed. Fix it, switch the group off, and the actual image is clean. Turn the viewing layers off before exporting; leaving them on is a classic and embarrassing mistake.

The limits

It does not create data. Dodging a black shadow lifts it to a noisy grey, because there was nothing there to reveal. Burning a clipped highlight gives you grey with no texture. Check what is actually in the area before spending twenty minutes on it.

It amplifies noise in the shadows. Lifting a dark region multiplies its noise along with its signal. Denoise first if the file is noisy, or restrict your dodging to areas with real signal.

It bands in 8-bit. Very gentle work over a smooth gradient — a studio backdrop, a sky — lands between the available levels and shows as terracing. This is a 16-bit job.

And it is slow. Properly dodging and burning a portrait is 20 to 45 minutes of work. It is why high-end retouching costs what it does, and why it is not the right tool for 400 catalogue images, where a masked curve is.

BEFORE YOU MOVE ON

A retoucher dodges and burns a portrait on a 50 per cent grey Soft Light layer and the result looks smooth on screen. Printed at A3, the cheek shows uneven patches. What check would have caught this, and why does it work?

A. Zooming to 100 per cent, which reveals per-pixel variation that fit-to-screen viewing averages away.

B. Switching the layer from Soft Light to Overlay temporarily, which doubles the effect and makes errors visible.

C. A temporary Black and White layer under a very steep S-curve at the top of the stack, which exaggerates small tonal differences until the unevenness is obvious.

D. Checking the histogram for comb-like gaps, which indicate the dodging has exhausted the available levels.

Why Clarity makes halos, and what Dehaze is estimating

THE ONE THING TO KEEP

Shadows/Highlights, Clarity, Texture and sharpening are one operation at four radii — comparing each pixel to a blurred version of its surroundings — so halos are inherent to the method, while Dehaze is a separate estimate of an atmospheric veil that fails wherever the scene genuinely has no dark pixels.

One mechanism behind four sliders

Shadows/Highlights, Clarity, Texture, Structure and Dehaze look like five separate features. Four of them are the same operation at different scales.

The operation: make a **blurred copy** of the image's luminance. Compare each pixel to its blurred version. That comparison tells you whether the pixel is brighter or darker than its own surroundings — in other words, it separates *local* contrast from *global* brightness. Then adjust based on that difference.

- **Shadows/Highlights** uses a large blur radius and applies a brightening or darkening that depends on the local average, so a dark pixel in a dark region gets lifted while a dark pixel next to a bright one does not.
- **Clarity** uses a medium radius and increases the difference, adding contrast at the scale of features rather than edges.
- **Texture** uses a small radius, so it affects fine detail — skin pores, fabric weave — without touching broader tonal shapes.
- **Unsharp masking** is the same operation at a radius of one pixel or less, which is the sharpening lesson later.

Once you see it as one mechanism with a radius control, the sliders stop being mysterious and the artefacts become predictable.

The halo is the mechanism, not a bug

A halo appears wherever there is a strong edge. Consider a dark tree against a bright sky. Near that boundary, the blurred copy is a mixture of tree and sky. So a sky pixel close to the tree is compared against a local average that is darker than the sky, and the tool concludes it is a *bright* pixel and brightens it further. A tree pixel near the sky is compared against an average that is brighter, so it gets darkened.

The result is a bright band in the sky along the branches and a dark band inside them. That is the halo. It is a direct consequence of using a blurred neighbourhood as the reference, and every implementation has it to some degree.

The defences, in order of usefulness:

1. **Less.** The single most effective control. Clarity at +15 is invisible; at +60 every edge rings.
2. **A smaller radius**, where the tool exposes one. The halo shrinks with the blur.
3. **Masking.** Apply local contrast through a mask that excludes the high-contrast boundaries, or apply it on a duplicate layer with a mask.
4. **Better algorithms.** Edge-aware filters — bilateral, guided and local Laplacian — weight the blur so pixels across an edge contribute less. This reduces halos substantially. darktable's *local contrast* module offers both a bilateral and a local Laplacian mode; the Laplacian one is noticeably cleaner on architecture.

Shadows/Highlights: the settings people never open

The default Shadows/Highlights in Photoshop is aggressive and produces the grey, flat, obviously-processed look that gives the tool its bad reputation. Click **Show More Options** and three controls appear that make it usable:

- **Amount** — how much lifting. Start at half the default.
- **Tone** (older versions: Tonal Width) — how far up the tonal range the effect reaches. Narrow it so only genuine shadows are lifted and the midtones are left alone.

- **Radius** — the blur size. Too small and you get halos round every small object; too large and the effect becomes a global brightness change. Somewhere around a tenth of the image's short side is a reasonable starting point.

GIMP has *Colours > Shadows-Highlights* with the same three concepts. darktable's *shadows and highlights* module and its newer *tone equaliser* both do this, and the tone equaliser is the better tool: it splits the image into tonal bands by a blurred luminance estimate and lets you lift or drop each band with a mask preview showing exactly which pixels each band contains.

Dehaze is a different animal

Dehaze is not local contrast. It is an estimate of atmospheric scattering.

The model: haze adds a veil of light proportional to distance, so distant objects are both brighter and less saturated than they should be. The algorithm estimates how much veil is present at each point — usually via the *dark channel prior*, the observation that in a haze-free outdoor photograph almost every small patch contains at least one very dark colour channel — and subtracts it, scaling contrast back up.

That is why Dehaze does things Clarity cannot: it restores saturation as well as contrast, and it does so more strongly in the distance than the foreground.

And why it fails where it does. Pushed hard, it produces dark, oversaturated skies with cyan blotches, because the veil estimate is wrong in regions that genuinely contain no dark pixels — a white wall, snow, fog that is the subject rather than an obstruction. Negative Dehaze adds a veil, which is a fast way to make a fake atmospheric distance layer in a composite.

GIMP has no Dehaze; darktable's *haze removal* module implements the dark channel prior and is free.

Where each one belongs

- **Shadows/Highlights** — early, in the raw developer, as a recovery step. It is fixing a capture problem.

- **Clarity and Texture** — mid-pipeline, after tone, before colour grading, usually masked.
- **Texture, negative**, is the gentlest skin smoothing available: it reduces fine detail without touching tonal shape, so it does not produce the plastic look that blur does.
- **Dehaze** — early, and sparingly, and check the sky.

The honest limit

All of these are contrast redistribution. They make existing differences more visible; they do not add information. A hazy photograph dehazed hard shows you the noise and the JPEG blocks in the distance, because those were the only variations present, and you have just amplified them.

The same is true of shadow lifting: the shadow contains signal plus noise, and lifting multiplies both. If the shadow was three stops under on a high-ISO file, lifting it produces a noisy grey mess, and that is a capture limit rather than an editing one. The fix is a second exposure, a reflector, or a fill light — which is the compositing block's exposure blending, and it is the honest answer far more often than a slider is.

BEFORE YOU MOVE ON

An editor raises Clarity on a photograph of bare trees against a bright sky, and a pale band appears in the sky along every branch. What is happening?

A. The sky is clipping locally where the added contrast pushes bright pixels past 255, producing a white outline.
B. Near an edge, the blurred reference is a mixture of tree and sky, so sky pixels are judged brighter than their neighbourhood and brightened further.
C. Clarity operates on the luminosity channel only, and the mismatch between the adjusted luminance and the unchanged colour shows as a band.
D. The image is 8-bit, so the added contrast has exhausted the available levels along the highest-contrast boundary.

35 · 8 min

Saturation is a slider that destroys detail quietly

THE ONE THING TO KEEP

Raising saturation pushes channels apart until the leading one clips and flattens a shaded surface into a single colour, which is why Vibrance weights the increase against existing saturation and why the three-channel histogram and a gamut warning are the two checks that catch the damage.

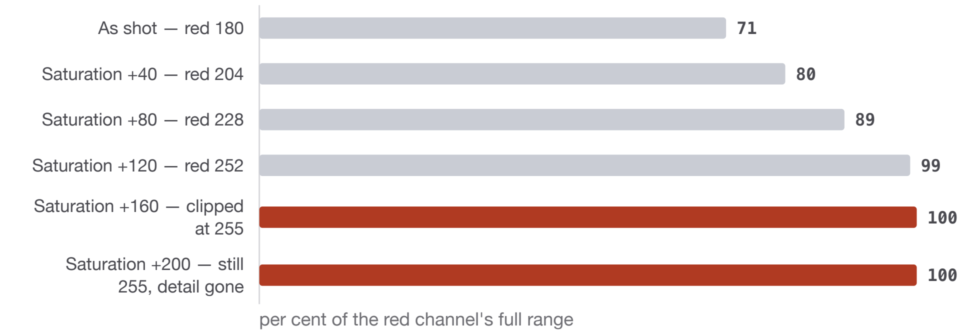
What raising saturation actually does

Saturation is distance from grey. Raising it pushes the three channel values further apart from their common average.

Take a pixel at 180, 120, 100 — a warm skin tone. Raise saturation by 40 per cent and it becomes roughly 204, 108, 84: the high channel goes higher, the low ones go lower. Do it again and the red channel reaches 255, and stops.

That stopping point is where the damage happens. Once a channel clips, every pixel above the clipping point in that channel becomes the same value. A red rose that had shading, texture and a gradient across each petal becomes a flat red shape. This is the mechanism and it is completely silent — there is no warning, and on many screens the flat shape looks *more* impressive than the original, which is why over-saturated images are so common.

One warm skin pixel, red channel, saturation pushed five times



The pixel starts at 180, 120, 100. Each push moves the channels further apart until the leading one reaches 255 and stops. Every pixel past that point becomes the same value, so a petal that had shading across it becomes a flat shape — silently, with no warning anywhere except the three-channel histogram.

The three-channel histogram from earlier in this block is the check. Raise saturation, look at the histogram, and if a channel is piling up against 255 you have gone past the point of no return.

Vibrance, and what it actually protects

Vibrance applies a saturation increase that is **inversely weighted by how saturated the pixel already is**. Already-vivid colours barely move; muted colours move a lot.

It also carries a specific protection for tones in the skin range — orange-yellow hues at moderate saturation get less of the increase than other hues. This is not a subtle piece of intelligence; it is a weighting curve, and it is why a Vibrance boost on a portrait against foliage greens the leaves without turning the face orange.

The practical rule: **Vibrance for photographs of people or mixed scenes, Saturation for a targeted push on a specific colour**. And for both, the useful range is smaller than you think. +10 to +20 Vibrance is visible; +40 looks processed.

Hue/Saturation is really six tools

The dropdown that says Master also contains Reds, Yellows, Greens, Cyans, Blues and Magentas. Choose one and the adjustment applies only to that band of hues, with adjustable falloff shown by the two pairs of markers on the colour bars at the bottom of the dialogue.

Those markers are the part nobody touches and they matter. The inner pair defines the range affected fully; the outer pair defines how far the effect fades out. Dragging the outer markers wide gives a gentle transition; keeping them close to the inner ones gives a hard cut-off and a visible edge where the adjustment stops.

The **targeted adjustment tool** — the small hand icon — is faster than any of it. Click it, then drag on the photograph itself: dragging left and right on a yellow sari shifts its hue, and the dialogue works out which band you meant.

Uses that come up daily:

- **Yellows, saturation down** — fixes the greenish-yellow cast that fluorescent tubes leave on skin.
- **Blues, luminance down** — deepens a sky without a mask and without touching anything else.
- **Reds, hue shifted slightly towards orange** — corrects the over-red skin that heavy sharpening and contrast produce.
- **One product colour shifted** to show a second colourway.

Where the free tools sit

GIMP's *Colours > Hue-Saturation* has the same six ranges and an overlap control that does what the falloff markers do. Its *Colours > Hue-Chroma* works in a perceptual space (LCh), which is genuinely better behaved: raising chroma there changes colour purity without dragging lightness with it, which the RGB-based saturation control cannot avoid doing.

darktable's *colour balance RGB* module has separate global, shadow, midtone and highlight chroma controls plus a vibrance equivalent, and it works in a perceptual space by default. For anyone doing serious colour work this is better than anything in Photoshop, and it is free.

Krita and Photopea both have standard Hue/Saturation implementations.

The gamut trap on export

Here is the consequence that surprises people. Saturation raised inside a wide working space produces colours that are perfectly valid there and have no representation in sRGB or in CMYK.

They look fine while you edit. On conversion to sRGB for the web, they clip to the boundary — several distinct colours collapsing onto one, exactly as the gamut lesson described. On conversion to CMYK for print, the same, worse, because CMYK's gamut is smaller again in the bright saturated corners.

So the file that looked vivid on your screen prints as a flat patch, and the client says the brochure does not match the website. The check is *View > Gamut Warning* with the destination profile loaded, before you commit. Ten seconds.

The judgement, not the slider

The honest observation: saturation is the control that most reliably makes an inexperienced editor's work identifiable. Screens are bright, saturated colour is immediately pleasing, and the effect is cumulative — a Vibrance boost in the raw developer, a saturation boost in the editor, and a phone's own display saturation stack up into something garish that looked reasonable at every individual step.

Two habits that help. Make saturation moves on an adjustment layer so you can toggle them, and look at the before and after rather than only the after. And judge on the least saturated screen you own rather than the most, because the picture will look more saturated everywhere else, never less.

BEFORE YOU MOVE ON

A brand's bright orange packaging looks vivid on screen after a saturation boost but prints as a flat patch with no visible shading. What happened, and where should the problem have been caught?

A. The image was in 8-bit, so the orange had too few levels to hold shading, and working in 16-bit would have preserved it.
B. The printer's ink limit was exceeded, which is a press setting rather than anything visible in the file before delivery.
C. The saturated orange fell outside the CMYK gamut, so distinct shades clipped to the same reproducible colour, and a soft proof with a gamut warning would have shown it.
D. The file was delivered in sRGB rather than Adobe RGB, and Adobe RGB would have carried the orange through to press intact.

36 · 9 min

Monochrome is a decision about which colours become which greys

THE ONE THING TO KEEP

Monochrome conversion is a choice of how each hue maps to a grey, so the six-channel Black and White adjustment reproduces the old coloured lens filters — and a photograph whose subject and background differ in hue but not in lightness cannot be separated by any conversion.

Desaturate is the one method that throws the decision away

Dragging saturation to −100 collapses every colour to a grey determined by its lightness alone. Two objects of different colours but similar lightness become the same grey, and the picture goes flat. That flatness is what people mean when they say their black and white conversions look dull.

The realisation that fixes it: in monochrome, you choose the mapping. A red flower can be near-white or near-black in the same photograph, depending on how you weight the channels, and both are legitimate. Colour photography records what was there; monochrome conversion is an interpretation, and the interpretation is the work.

The Black and White adjustment, and the filter it imitates

A Black and White adjustment layer gives you six sliders — reds, yellows, greens, cyans, blues, magentas — each controlling how bright that hue becomes in the result.

This is a digital version of a hundred-year-old technique. Film photographers screwed coloured glass filters onto the lens:

- **A red filter** passes red and blocks blue, so a blue sky records as very dark and clouds leap out white. Every dramatic Ansel Adams sky is a red or deep-yellow filter.
- **A yellow filter** is the mild version, the standard landscape default.
- **A green filter** lightens foliage and is traditional for portraits of men, because it darkens the reds in skin and emphasises texture.
- **An orange filter** lightens skin and suppresses freckles and blemishes, which is why it was used for portraits of women in the studio era.

The sliders do the same thing after the fact, and better, because you have all six at once rather than one piece of glass.

The targeted adjustment tool works here too: click it and drag directly on the sky to darken it, on the skin to lighten it. Far faster than guessing which slider governs a given object.

The trap: darkening a channel to nothing

Pull the Blues slider hard negative to get a dramatic sky, and watch the sky's noise. You are amplifying the difference between the channels, and the blue channel is the noisiest. A strong negative blue on a high-ISO photograph produces a grainy, blotchy sky.

The same move on an 8-bit JPEG produces banding, because you are stretching a channel with 256 levels across a wider range of greys. Black and white conversion benefits from 16-bit more than almost any other operation, and from starting in a raw developer where the channels still hold their full precision.

Contrast in monochrome is the whole subject

Without colour, the only thing separating the subject from the background is tone. A photograph where the subject and the background happen to be different colours at similar lightness will not work in black and white no matter how it is converted, because there is nothing to separate.

So the judgement runs the other way from colour work: **decide whether the photograph has tonal separation before converting**, not after. A quick test is to add a Black and White layer with default settings and look — if the subject vanishes into the background, either one of the six sliders can pull them apart, or the photograph is a colour photograph.

What does work in monochrome: strong shapes, texture, directional light, faces, weather, architecture, anything where the interest is in form rather than hue. What does not: a picture whose point was that the sari was that particular green.

After the conversion

A monochrome image usually needs *more* contrast work than a colour one, because it has lost a dimension of separation and has to make it up in tone.

- **A stronger S-curve** than you would apply in colour, placed above the Black and White layer.
- **Dodging and burning**, which matters more here than anywhere else. The whole tradition of fine printing is local tonal control on monochrome.
- **Local contrast** on textures — stone, skin, fabric.

Toning, and why prints were never neutral

A pure neutral grey print is a modern idea. Silver prints were routinely toned — selenium for a cool purple-black and archival permanence, sepia for warm brown, gold for blue-black. Platinum prints are warm by nature.

Digitally: a Colour Balance or Curves layer above the conversion, shifting the shadows one way and the highlights the other. Warm highlights with cool shadows is the classic split, and a very small amount — five to eight points — is enough. More than that and it stops being a tone and starts being a colour cast.

GIMP does this with *Colours > Colour Balance* after a *Colours > Desaturate > Mono Mixer*, or with a Curves adjustment on individual channels. darktable has a dedicated *monochrome* and *colour balance RGB* pair. Krita has a Channel Mixer filter layer.

The free paths

- **GIMP**: *Colours > Desaturate > Mono Mixer* gives you channel weights; *Desaturate > Desaturate* offers Luminance, Lightness, Average and Value modes, and Luminance is the correctly weighted one.
- **darktable**: the *monochrome* module plus *colour calibration* in its grey mode, which lets you set channel weights precisely.
- **RawTherapee**: a black-and-white tool with channel mixer and preset film-filter emulations.
- **Photopea**: a Black and White adjustment layer matching Photoshop's.
- **Snapseed** on a phone has a Black and White filter with a colour-filter control, which is the same six-slider idea in simplified form.

The honest part

Converting to black and white does not rescue a weak photograph. It is a widespread belief — that a picture which is not working in colour might work in monochrome — and it is true occasionally, when the problem was a distracting colour, and false most of the time, when the problem was the picture.

What monochrome does is remove a variable, which makes the remaining ones — light, shape, moment, texture — carry everything. That is why it is a demanding discipline rather than a filter, and why photographers who work in it mostly decide before they press the shutter rather than after.

BEFORE YOU MOVE ON

A photographer converts a landscape to black and white and pulls the Blues slider strongly negative to darken the sky. The sky becomes dramatic but visibly blotchy and grainy. What is the mechanism?

A. Negative slider values invert the channel, and inverted channels carry double the noise of positive ones.
B. Darkening a hue reduces the exposure of that area, and underexposed regions are always noisier than correctly exposed ones.
C. A strong weight on the blue channel amplifies that channel's contribution, and blue is typically the noisiest channel in a photograph.
D. The Black and White adjustment works in 8-bit internally regardless of document depth, so any strong slider move quantises the result.

---

37 · 9 min

A look is a mapping, and a LUT is a table of it

THE ONE THING TO KEEP

A grade is a mapping from input colour to output colour, so a LUT applies it identically every time and therefore gives a different result on an image whose exposure and contrast were not normalised first.

Grading is not correction

Colour correction makes the picture neutral: white is white, skin is plausible, the cast is gone. **Grading** is what you do after that, deliberately moving the colour away from neutral for effect.

The order matters. Grade before you correct and you are building a look on top of an error, and it will not transfer to the next frame. Correct first, always, so you know where neutral is.

The three-way split

The classic grading control splits the tonal range into shadows, midtones and highlights and lets you push each one towards a hue. Photoshop's *Colour Balance* does it with three sets of sliders; Lightroom's *Colour Grading* panel and darktable's *colour balance RGB* do it with three colour wheels.

The move that underlies most cinematic grading is simple: **cool shadows, warm highlights**. Push the shadows towards blue-teal and the highlights towards orange-yellow. The reason it works is that skin is warm and most shadow is ambient skylight, which is genuinely blue — so the grade exaggerates a relationship the eye already expects, and it separates the subject from the background by hue as well as tone.

The inverse — warm shadows, cool highlights — reads as period, faded, or like an old print. Both are worth a few minutes of experiment.

Doing the same thing with **Curves** gives more control: switch to the Blue channel, lift the bottom left corner slightly and drop the top right slightly, and you have exactly the teal-and-orange split with control over where it transitions. This is worth learning because it works in every tool including free ones.

What a LUT is

A lookup table is exactly what the name says: a table saying, for a given input colour, what the output colour should be.

A 3D LUT is a cube. A common size is 33×33×33, which is 35,937 entries covering the RGB space at intervals; colours between the entries are interpolated. A `.cube` file is a plain text file listing those entries, and you can open one in a text editor and read it.

```
TITLE "teal and orange"
LUT_3D_SIZE 33
0.000000 0.000000 0.000000
0.031250 0.028000 0.040000
...
```

Because it is just a mapping, a LUT can encode anything a combination of curves, saturation moves and channel mixes can do — and it applies in one operation, identically, every time.

Apply one in Photoshop with a **Colour Lookup** adjustment layer. GIMP has *Colours* > *Map* > *Apply LUT* in recent versions and the GEGL operation behind it. darktable has a *LUT 3D* module. Krita, Resolve, Premiere, Kdenlive and Shotcut all load `.cube` files, which is why a look built for video can be used on stills.

Making your own

The useful workflow, and it takes ten minutes.

1. Grade one representative image until you like it, using adjustment layers only.
2. In Photoshop, *File* > *Export* > *Colour Lookup Tables*. It renders your whole adjustment stack into a `.cube`.
3. That file now applies the same look to anything, in any application that reads LUTs, including free video editors.

Without Photoshop, the route is a **HALD image**: a small PNG containing every colour in the space laid out in a grid. Apply your edits to the HALD image in GIMP, and the modified HALD *is* a LUT — GIMP's G'MIC plugin and ImageMagick both convert between HALD PNGs and `.cube` files. This is how most free film-emulation LUT packs were built.

The limit that makes LUTs go wrong

A LUT is a fixed mapping from input value to output value. It knows nothing about the photograph.

That means **two images at different exposures get different-looking results from the same LUT**, because their pixel values are different and the mapping is by value. A look built on a correctly exposed frame applied to one that is a stop darker will land on the wrong part of the curve and come out muddy or contrastless. This is the single commonest complaint about LUT packs and the answer is always the same: normalise exposure, contrast and white balance *first*, then apply the look.

The second limit is **clipping inside the LUT**. A strongly contrasty LUT can map several input values to the same output, destroying detail exactly as a hard curve does, and unlike a curve you cannot see the shape to know. Apply LUTs on an adjustment layer with opacity control, at 40 to 70 per cent, and check the histogram.

The third is **the input space**. A LUT built for log-encoded video footage applied to an ordinary sRGB photograph produces a wildly wrong result, because the mapping assumes an input distribution that is not there. Free LUT packs are often video LUTs; check what they were made for.

Film emulation, honestly

A lot of free and paid LUTs claim to be Portra, Tri-X, Velvia or Kodachrome. What a LUT can reproduce is the colour mapping — the hue shifts, the contrast curve, the way a film rendered greens or skin.

What it cannot reproduce: grain structure (which is spatial, not per-pixel), halation (light spreading inside the film base around highlights), the response to overexposure (film rolls off gently, digital clips hard), and the tonal response varying with exposure. Those are the parts that actually make film look like film.

The better free tools do model some of this. **RawTherapee's** film simulation uses HaldCLUTs; darktable has a *sigmoid* and *filmic rgb* module that models the highlight roll-off properly, which is the single most film-like thing you can do to a digital file, and it is not a LUT at all — it is a tone mapping curve with shoulder and toe controls.

A look built from your own decisions on your own photographs will serve you better than a pack of 300 named presets. Build three: one neutral, one warm, one cool. Use them for a year. That is a style, and a pack is not.

BEFORE YOU MOVE ON

A photographer applies the same purchased LUT to two frames from one shoot. On the correctly exposed frame it looks good; on a frame a stop darker it comes out muddy and flat. Why?

A. The LUT is a fixed mapping by pixel value, so a darker frame's values land on a different part of the table than the look was designed around.
B. LUTs are applied before the exposure adjustment in the processing pipeline, so the exposure correction is being undone.
C. The LUT's interpolation between table entries is less accurate at low values, so dark frames receive a cruder approximation.
D. The darker frame has more noise, and the LUT amplifies noise, which reads as muddiness.

---

38 · 10 min

There is no correct number for skin

THE ONE THING TO KEEP

Skin has no universal correct value, so the method is to neutralise from a known reference, check hue consistency across the face rather than against a target, and recognise that automatic exposure and white balance were tuned on distributions that systematically under-serve darker skin.

The rule you will be told, and its actual status

Search for skin tone correction and you will find a rule: in CMYK, yellow should be slightly higher than magenta, and cyan should be a fraction of magenta — often quoted as cyan being a quarter to a third of the magenta value, with magenta and yellow close together.

It is a genuine rule of thumb from the pre-press era, and it works within its range: it was derived from Caucasian skin printed on coated stock. Applied to darker skin, the ratios change substantially — cyan rises considerably relative to magenta, and the yellow-magenta relationship shifts. Applied as a universal target, it lightens and pinkens dark skin, which is exactly the failure mode this lesson is about.

The honest statement: **skin covers an enormous range of hues, lightnesses and saturations, and there is no single set of numbers that is correct.** What there is instead is a method.

The method that works

1. Get neutral first. Set white balance from something known to be neutral in the frame, as in the earlier lesson. Most "the skin looks wrong" problems are white balance problems, and fixing skin directly while the whole frame has a cast means fighting the same error twice.

2. Check that the cast is gone from something that is not skin. A white shirt, a grey wall, the whites of the eyes. If those are neutral and the skin still looks wrong, the problem is in the skin.

3. Look for hue consistency across the face rather than a target value. Skin should be one hue family across the whole face, varying in lightness and saturation. The commonest genuine fault is a **hue split** — the forehead one hue, the cheeks another, the jaw a third — caused by mixed lighting, makeup, or a heavy-handed adjustment. Sample five points with a 5×5 average eyedropper and compare.

4. Compare against something real. A reference photograph of the same person under known light. A colour checker in the first frame. The person's own description. Nothing else is evidence.

5. Judge at the size it will be seen, against a neutral grey background rather than a white one. A face against white looks darker and more saturated than the same face against grey.

The history, because it explains the tools

This matters for understanding why automatic tools behave as they do.

Colour film stocks and printing processes were calibrated for decades against reference cards — the "Shirley cards" — showing a white woman. Laboratory printing equipment was set up so that card came out correctly, which meant the entire chain was optimised for light skin. Darker skin was reproduced poorly, with detail lost in the shadows, and this was a well-documented, widely-experienced failure rather than an occasional one.

Kodak made significant improvements to the shadow response of its colour stocks in the 1970s and 80s. The documented drivers included complaints from furniture and chocolate manufacturers, who could not get dark wood and dark chocolate to reproduce, as much as from photographers of people. Multiracial reference cards appeared later.

The pattern repeated in digital. Automatic white balance, automatic exposure and face detection are all built by measuring performance against datasets, and where those datasets have under-represented darker skin, the resulting systems have performed worse on it. This has been measured repeatedly in published audits of commercial face-analysis systems, with error rates differing substantially across skin tone and gender groups.

The practical consequence for you at the editing desk: **automatic exposure, automatic white balance, automatic "portrait" modes and one-click skin tools were tuned on a distribution, and if your subject sits outside it, the automatic answer will be systematically wrong in a consistent direction.** Knowing that turns a vague dissatisfaction into a specific correction.

Practical corrections that do not flatten

Exposure for dark skin. The commonest error is underexposure, because the camera's meter reads a dark subject and pulls the exposure down. Expose for the skin, not the average — spot meter on the face, or simply add a stop and check the histogram. Lifting underexposed dark skin in post brings up noise and loses the specular highlights that give skin its life.

Do not chase a target lightness. Lightening dark skin to match a reference of light skin is not a correction; it is an alteration of the person. The same applies to the automatic "beautify" modes in phone cameras that lighten and smooth by default, which many phones apply without asking.

Saturation, carefully. Dark skin often carries more colour variation than a global saturation slider handles well, and pushing saturation globally shifts it towards red. A targeted Hue/Saturation on Reds and Yellows, or better, a perceptual chroma control like GIMP's Hue-Chroma or darktable's colour balance RGB, keeps lightness stable.

Specular highlights matter more, not less. On darker skin the highlights carry much of the form. Crushing them with a heavy highlight recovery flattens the face. Keep them and control them locally with burning instead.

Mixed skin tones in one frame — a family group, a team photograph — cannot all be optimised globally. Correct globally for neutral, then use masked local adjustments per person. This is normal professional practice and it is not a failure.

What to do when you are unsure

Ask. If the photograph is of a person and you can reach them, show them two versions. This is not a cop-out; it is the only source of ground truth about how somebody's own skin looks, and it takes one message.

For commercial work, put a colour checker in the first frame of each setup. It converts an argument about taste into a measurement, and it costs about the price of a lens cap.

What remains genuinely unresolved

There is no agreed standard for "correct" skin reproduction across the full human range, and the industry references that exist were built for particular media and particular populations. Different display technologies, different printing processes and different viewing conditions all shift the answer.

The defensible position for an editor is procedural rather than numerical: neutralise from a known reference, keep hue consistent across the subject, preserve rather than flatten the tonal range, compare against a real reference where one exists, and be explicit about what you changed. That is a standard you can state and be held to, which is more than a set of CMYK ratios can offer.

BEFORE YOU MOVE ON

An editor corrects a portrait of a dark-skinned subject by adjusting until the skin's CMYK values match a widely-quoted ratio derived from pre-press practice. The subject says it does not look like them. What is the most accurate assessment?

A. The ratio applies to print only, so the on-screen version will be correct once converted to CMYK for output.

B. The subject is judging on an uncalibrated screen, so their impression is not reliable evidence about the correction.

C. The ratio was derived from light skin on coated stock and does not describe the full range of human skin, so matching it has altered the subject rather than corrected the file.

D. The white balance must still be wrong, because a correctly neutralised file would naturally land on the standard ratio.

39 · 8 min

Forty photographs that have to look like one shoot

THE ONE THING TO KEEP

Set consistency is bought at capture with a grey card and manual settings, maintained by syncing one hero frame's settings and then adjusting only exposure, and verified on a mid-grey contact sheet with sampled numbers rather than by eye.

The failure is visible even when each file is fine

A catalogue, a menu, a property listing, a wedding album, a shop's product grid. Each image can be individually well-edited and the set can still look wrong, because the eye compares neighbours ruthlessly. A grid where one white background is 252 and the next is 247 reads as sloppy instantly, though neither is wrong on its own.

Consistency across a set is a separate skill from editing one image, and it is mostly an organisational skill rather than a technical one.

Forty photographs that have to look like one shoot

Before the first frame	Manual exposure and manual white balance. Automatic modes re-decide per frame, so a dark product and a light one come back at different exposures under identical light.
First frame of each setup	One frame of a grey card. Five seconds, and white balance stops being an opinion. Shoot a new one if the lights move.
In the developer	Edit the hero frame fully, then paste its whole history stack onto every other frame from that setup.
Frame by frame	Adjust exposure only. Re-shape the tone curve per frame and the set drifts apart again, one small reasonable decision at a time.
Before delivery	Contact sheet on mid-grey, never white, and sample the background in three corners with a 5 by 5 average. White hides exactly the difference you are hunting.

Consistency is bought at capture and maintained by refusing to re-decide. The eye compares neighbours ruthlessly: a grid where one background is 252 and the next is 247 reads as sloppy instantly, although neither is wrong on its own.

Anchor the shoot before you edit it

The cheapest consistency comes from capture.

- **A grey card or colour checker in the first frame of every lighting setup.** One frame. It costs five seconds and it makes white balance a measurement.
- **Manual exposure and manual white balance** on the camera, not automatic. Automatic modes re-decide per frame, so a dark product and a light product shot under identical light come back at different exposures and different casts, and you spend an hour undoing it.
- **Do not move the lights between products.** If you must, shoot a new grey card frame.
- **Note what changed.** A short text file — `setup A: frames 1–60, window + white card. setup B: frames 61–120, moved to the table.` Future you will need it.

Pick a hero frame and work outward

The method:

1. Choose the frame you most care about — the one going on the homepage — and edit it fully.
2. Copy its develop settings to every other frame from the same setup. In darktable this is *copy/paste history stack*; in RawTherapee, copy and paste the processing profile; in Lightroom, *Sync Settings*; in Photoshop's Camera Raw, select all and synchronise.
3. Then go through the set individually and adjust only what differs — usually exposure by small amounts, because the product's own lightness changed.
4. Finish by comparing, not by editing.

Step 3 has a discipline attached: **adjust exposure, not the tone curve.** If you re-shape the curve per frame, the set drifts apart. Keep the curve identical across the set and move the exposure slider, which shifts everything uniformly.

Comparing properly

The eye is good at comparison and bad at memory, so put the images next to each other.

A contact sheet. Build one from the finished exports and look at the whole set at once.

```
magick montage ./final/*.jpg -tile 8x -geometry 220x220+6+6 -
background "#808080" check.jpg
```

Use a mid-grey background rather than white — white hides the difference between a 250 background and a 255 one, which is precisely the difference you are hunting.

Numbers where the eye is unreliable. Sample the same feature on several frames with a 5×5 average eyedropper and compare. For a catalogue on white, sample the background in three corners of each image; they should be within a couple of units. For a set of portraits, sample the forehead.

Two at a time, side by side, at the same size. Most developers have a compare view. Flipping between two images in the same window is better still, because a change shows as movement.

What cannot be matched

Light that changed during the shoot. An outdoor set shot across four hours has genuinely different light in the first and last frames — different colour, different direction, different contrast. You can reduce the difference; you cannot remove it, because the shadows fall in different directions and no colour adjustment moves a shadow.

Different cameras. Two bodies, even from the same manufacturer, render colour differently. Matching them means profiling both with a colour checker and building a correction for one, which is a real job and worth doing for a recurring client.

Mixed capture formats. Half the set raw and half JPEG will never match precisely, because the JPEGs have the camera's tone curve baked in and the raws do not.

Images from different sources — a client's old photographs mixed with your new ones. Here you are matching *towards* something rather than matching a set, and the honest conversation is about how close is achievable. Often the answer is to reshoot the old ones.

A short checklist before a set goes out

- Same pixel dimensions and aspect, or a deliberate and documented set of them.
- Same background value, measured, not eyeballed.
- Same product scale within the frame.
- Shadows present on all or on none.
- Same colour profile embedded on every file.
- Same naming pattern, machine-readable.
- Contact sheet reviewed on grey.
- Three files opened at 100 per cent, chosen to be the hardest cases.

That checklist is ten minutes and it is the difference between a set that looks commissioned and one that looks assembled. It is also the single most common reason a first client comes back with more work, which is worth more than any individual retouching skill in this course.

BEFORE YOU MOVE ON

An editor is matching forty product shots. For each frame they adjust the tone curve individually until the product looks right, keeping exposure fixed. The set still looks inconsistent. Why is this approach working against them?

A. Tone curves are applied after white balance, so each per-frame curve reintroduces a different colour cast.
B. Reshaping the curve per frame changes the relationship between tones differently in each image, whereas an exposure move shifts everything uniformly and preserves the shared look.
C. Forty individual curves exceed what can be synchronised between frames, so the settings cannot be copied and checked.
D. Curves operate on 8-bit values, so repeated per-frame adjustments accumulate quantisation differences across the set.

CASE STUDY · MODULE 4

One hundred and eighty graduates, one corridor, and a camera that kept deciding

A degree college in Hyderabad photographing its graduating year in a corridor between the library and a window, on the one day everybody was on campus.

The setup, and what was wrong with it

A borrowed camera on a tripod, a maroon cloth backdrop taped to a noticeboard, six-minute slots, and a student from the photography club operating it. Daylight came through a window on the left. Fluorescent tubes ran along the ceiling. Everything was on automatic: automatic exposure, automatic white balance, and the camera shooting JPEG because nobody had thought about the card filling up.

By the time a lecturer looked at the files at lunch, sixty-three students had been photographed and two problems were visible on the laptop screen.

The exposures were not consistent, and they were not randomly inconsistent. The camera's meter reads the frame and pulls towards a middle value. A student with dark skin fills much of the frame with a dark subject, so the meter concluded the scene was under-lit and opened up — or, in the sequence where the backdrop dominated, did the opposite and pulled down. Across the sixty-three, the darker-skinned students were on average noticeably under-exposed, and the failure was systematic rather than scattered.

The backdrop wandered. Automatic white balance re-decides per frame. A frame with a lot of maroon in it reads as a colour cast and gets corrected towards neutral, taking some of the maroon with it. Sampled with a 5 by 5 average, the backdrop measured between 243 and 251 across the set and carried a slight green shift on the frames where the tubes had dominated over the window.

No individual photograph looked wrong. The set, laid out as a contact sheet, looked assembled rather than commissioned.

The decision at lunch

Stop, go manual, reshoot the sixty-three. Correct, and forty-one of those students had already left campus, several of them travelling home that evening and not returning before the prints were needed.

Carry on and fix it in the developer. Nine hours of per-frame work, and the arithmetic of the under-exposed frames is unkind. Lifting exposure afterwards multiplies the signal and the noise together, and on an 8-bit JPEG it does so inside 256 levels. On darker skin, where much of the form is carried by the specular highlights, a heavy global lift also flattens the face — which is a worse outcome than the under-exposure it is fixing.

Switch to manual now and match the first sixty-three towards the second set. Cheapest, and it accepts two sets that a contact sheet will always be able to tell apart.

The cost was on both sides and it was not only technical. Asking forty-one students to come back is asking forty-one people for an hour of a week they had already finished.

The decision made three weeks earlier

There was a fourth cost, and it had been incurred before anybody picked up the camera. The club had set it to shoot JPEG so the card would not fill.

In a raw file, white balance is a pair of multipliers that have not been applied yet, so changing it costs nothing at all. In a JPEG it has been baked in, and correcting it means stretching one channel against the others inside 256 levels. Correcting a green fluorescent cast in a JPEG means multiplying part of the green channel down and the others up; a channel stretched that far bands in the smooth areas and goes noisy in the dark ones, and in the deepest shadows there is nothing left to stretch.

The same argument applies twice over to exposure. A raw file at 12 or 14 bits carries levels in the shadows that an 8-bit gamma-encoded JPEG simply does not have, which is exactly the region the under-exposed frames needed lifting from.

So the frames that most needed correcting were the frames least able to take it, and the reason was a setting chosen for storage on a day when storage was not actually the constraint.

> **STOP HERE**
>
> Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

They switched to manual exposure and manual white balance after lunch, shot a grey card frame at the start of each of the three remaining lighting changes, and put a notice on the department board that anyone photographed before half past twelve could come back on Monday morning. Twenty-two returned.

The remaining forty-one were corrected individually rather than in a batch. White balance came from the whites of a shirt or from the backdrop where no shirt was available; exposure was raised in the developer; and the specular highlights on the face were deliberately left alone rather than recovered, because on those frames the highlights were most of what gave the face shape. The corrections were done as masked local adjustments per student, not as one global move, which is ordinary professional practice and not an admission of failure.

The prints went out on time. Put side by side, the two sets are distinguishable, and the mid-grey contact sheet is where that shows — which is why the department now builds one before anything goes to the printer.

Two rules were written into the photography club's one-page sheet afterwards. Expose for the skin, not for the average. And never let the camera decide the same thing twice.

Why did the meter underexpose some students and not others, and what is the fix at capture rather than in post?

A camera's meter has no idea what it is looking at; it measures the light reaching it and pulls the exposure towards a middle value. A frame filled with a darker subject measures lower, so the automatic system moves the exposure in a direction that is wrong for that subject, consistently and in the same direction every time. The failure is systematic rather than random, which is exactly what makes it invisible frame by frame. The fix at capture is manual exposure set for the skin rather than the average — spot meter on the face, or add a stop and check the histogram — and then not letting the camera re-decide between frames.

Why is lifting the exposure afterwards not equivalent to having exposed correctly?

Because a lift multiplies whatever is in the shadow, and what is in the shadow is signal plus noise. A correctly exposed frame carries more signal relative to the noise floor to begin with. On an 8-bit JPEG the problem compounds: the stretch spreads 256 levels over a wider range and leaves gaps, which show as banding and blotches in the smoother areas of skin. There is a second cost specific to this job — a heavy global lift also pulls up the specular highlights that carry much of the form on darker skin, so the face flattens. That is why the corrections here were local and masked rather than global.

No single frame was wrong, and the set still looked wrong. What was the measurement that showed it, and why did it have to be made on grey?

Sampling the backdrop with a 5 by 5 average eyedropper on several frames and comparing the numbers: between 243 and 251 across the set, with a green shift on some. The eye is very good at comparison and very bad at memory, so the difference is invisible one file at a time and obvious in a grid. The grid has to be laid out on mid-grey rather than white, because white hides exactly the difference being hunted — a 250 background and a 255 background both read as white against a white page, and the whole question is whether they match.

MODULE 4 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A standard S-curve makes the picture punchier and costs you detail in the highlights and shadows. Where did the extra midtone contrast come from?

A. From the ends of the curve, which are now flatter
B. From the saturation increase the curve applied at the same time
C. From the 8-bit file gaining levels as the curve stretched it
D. From the black and white points, which the curve moved inward

2 A photograph of a deep red flower looks correctly exposed on the luminosity histogram, and every petal has lost its internal shading. Which view would have shown the problem?

A. The waveform, which keeps horizontal position
B. The camera's own histogram on the back of the body
C. The three channels drawn together
D. Photoshop's combined RGB histogram from the menu

3 An office interior has a green cast on everybody's faces, and the temperature slider only makes it bluer or more orange. What is the reason?

A. The white balance was baked in at capture and cannot be changed
B. Fluorescent tubes flicker, so the cast differs across the frame
C. The file must be converted to 16-bit before the cast will move
D. Green lies at right angles to the blue-yellow axis

4 Why do retouchers dodge and burn on a 50 per cent grey layer set to Soft Light rather than using the Dodge and Burn tools in the toolbar?

A. The toolbar tools only work on the background layer of a document
B. Mid-grey is neutral there, so nothing is destroyed
C. Soft Light is twice the strength of the toolbar tools
D. The toolbar tools cannot be used with a tablet's pressure curve

5 Clarity at +60 puts a bright band in the sky along every branch and a dark band inside the branches. What is producing the bands?

A. The sky is clipping where the adjustment pushed it past 255
B. Clarity raises saturation, and the sky's blue is out of gamut
C. Each pixel is compared to a blur of its area
D. The JPEG's 8 by 8 blocks are being exaggerated along the edge

6 You raise saturation until a red rose looks vivid on screen, and each petal becomes a single flat shape. What has happened inside the file?

A. The red channel reached 255 and stopped
B. The image was converted to a smaller colour space during the adjustment
C. The hue rotated past the edge of the red band in Hue/Saturation
D. The adjustment layer was set to Colour rather than Normal blend mode

MODULE 5

Repair and retouching

Removing what should not be there, and rebuilding what has been lost. This block separates texture from tone so skin can be evened without being sanded, works through portraits and products with the restraint that keeps them honest, deals with reflections and glare, reconstructs large removed objects rather than smearing them, scans and restores damaged prints, and ends with a plain account of what no tool can bring back.

> BY THE END OF THIS MODULE YOU CAN
>
> Repair a damaged or imperfect photograph to a professional standard using the right mechanism for each fault — separated frequencies, patch synthesis, manual reconstruction or warp — and state precisely which faults in a given file are unrecoverable and why

40 · 10 min

Three tools that all remove a spot, and they are not interchangeable

THE ONE THING TO KEEP

Healing blends into its surroundings, cloning copies exactly, content-aware borrows from elsewhere in the frame, and generative fill invents — and only the last one changes what the picture claims.

Four different machines with one apparent job

A mark on a wall, a cable across a floor, a spot on a face. Four tools remove them, and they work in completely different ways. Knowing which one to reach for is most of retouching.

Clone Stamp. You Alt/Option-click a source point, then paint. It copies pixels from source to destination exactly — same colour, same brightness, same texture, no blending. Total control, and it will leave an obvious repeated patch if you are careless.

Healing Brush and Spot Healing Brush. Same copying of texture, but then it discards the source's colour and brightness and mathematically blends the patch's edges into the surrounding tone. That is why healing a spot on a cheek works when cloning leaves a pale disc — the healing brush borrows the texture and inherits the local colour.

Content-Aware Fill / Patch. Select a region and the algorithm searches the *rest of the image* for patches that fit, then stitches them in. Everything it puts in was somewhere else in your photograph.

Generative Fill. A diffusion model generates new pixels from a text prompt and the surrounding context. It is not copying anything from your image. It is inventing.

The free landscape: GIMP has Heal and Clone tools that are genuinely fine, plus **Resynthesizer**, a plugin that does content-aware fill well and predates Adobe's. Krita has a *Smart Patch* tool. Photopea has both a healing brush and a clone stamp. None of the free tools has generative fill built in; you would generate separately and composite by hand, which is the workflow described in AI images.

Where each one fails, and why

The healing brush next to a strong edge. Heal a cable where a pale wall meets a dark table and you get a grey smear. The mechanism: the tool blends the patch's edges into what surrounds it, and what surrounds it here is both the wall and the table. It averages them. There is no setting that fixes this. Clone Stamp, sampling from along the same side of the edge, is the correct tool, and afterwards you can heal the middle of the patch.

The clone stamp on anything with visible structure. Clone a section of brickwork and the eye finds the repeated mortar joint instantly. Take several small samples from different places rather than one big one.

Two tools that both remove a spot, and are not interchangeable

	What it takes from the source	Where it fails
Healing brush	**Texture only** tone is inherited from the surroundings	**At a strong edge** it averages the pale wall and the dark table into a grey smear
Clone stamp	**Texture, tone and co** an exact copy, which is why it is right	**On visible structure** the repeated mortar joint is found by the eye immediately

Heal the middle of a cheek and clone anything near an edge — then heal the middle of the cloned patch to settle its tone. No setting rescues a heal that straddles a boundary, because averaging the two sides is what the tool is for.

Content-aware fill on something unique. Ask it to remove a person standing in front of a doorway and it will fill the doorway with more wall, because it can only use what exists elsewhere in the frame. There is no other doorway to borrow.

Generative fill on a real product. It will invent a plausible bottle cap that is not your client's bottle cap. Where the object must be the actual object, generate nothing — clone, patch, or reshoot.

Do it non-destructively

The trick that costs nothing: create a new empty layer above the photo and, in the tool options bar, tick **Sample All Layers** (Photoshop, Photopea) or *Sample merged* (GIMP). Now heal and clone onto the empty layer. The repairs live above the photograph, you can erase any one of them, and you can toggle the whole retouch off to compare. Every professional retoucher works this way.

For skin, the standard technique is **frequency separation** — splitting the image into a blurred layer holding colour and tone and a second holding pores and hair, so you can even out blotchy colour without sanding away the texture that makes skin look like skin. GIMP does it with *Wavelet decompose* or G'MIC; Photoshop and Photopea with a duplicate, a Gaussian blur and Apply Image. Learn it, because the alternative — blurring skin — is why so much retouching looks like plastic.

The line where retouching becomes a claim

This part is not a style question.

Removing dust on the sensor, a lens flare, a bin in the corner of a wedding photo, a scratch on a scanned family picture — nobody is misled. This is the darkroom work photographers have always done.

The line is crossed when the edit changes what the picture asserts about the world.

- **Documentary and news.** Reuters, AP and the major photojournalism competitions prohibit adding or removing content. Photographers lose their careers over a cloned-out sensor spot in a war photograph. If the image is presented as a record, the only permitted work is what a darkroom could have done: exposure, contrast, colour, crop.

- **Bodies.** Several countries now legislate this. France has required commercial images to carry a "photographie retouchée" notice where body shape has been altered since 2017; Norway passed a similar labelling requirement in 2021; Israel's law goes back to 2012. The direction of travel is one way.
- **Products.** Showing a product that does not look like the product is a consumer protection matter in most jurisdictions long before it is a design matter.
- **People's faces.** Consent for someone's likeness is not something you can retouch your way past. If you would not have photographed them doing it, do not generate them doing it.
- **Generated content.** If generative fill has invented an object, a person or a background that was never there, you have made synthetic media, and several countries — India's IT Rules on synthetically generated information among them — require that to be visibly labelled. The rules are recent and still moving. A lawyer in your country answers your specific case; the shape of it is in making things with AI.

A working test that survives every future tool: **would the person looking at this be surprised to learn what you did?** Dust removal, no. A removed person, yes. Between those sits your judgement, and writing down what you did in the delivery note costs nothing.

Today: remove one mark on a plain wall twice — once with the clone stamp, once with the spot healing brush, both on an empty layer. Then try the same mark sitting on an edge.

BEFORE YOU MOVE ON

A retoucher uses the Spot Healing Brush to remove a cable running exactly along the line where a pale wall meets a dark table. The result is a grey smear. What is the best diagnosis?

A. The brush was set to Content-Aware instead of Proximity Match; switching the mode would resolve it.
B. The image resolution is too low for the healing brush to find enough sample texture near the repair.
C. Healing blends the patch into its surroundings, and here the surroundings are both the wall and the table, so it averages them; the Clone Stamp sampling from one side of the edge is the right tool.
D. Only generative fill can reconstruct across a boundary like this, so the repair needs an AI tool.

41 · 10 min

Splitting texture from tone

THE ONE THING TO KEEP

Frequency separation splits an image into a blurred tone layer and a neutral-grey detail layer that recombine exactly, so blemishes are healed on the detail layer where there is no tone to get wrong and colour is evened on the tone layer where there is no texture to destroy.

The problem it solves

Skin has two kinds of imperfection and they need opposite treatments.

Tonal unevenness — a red patch on a cheek, a shadow under the eye, blotchy colour across a forehead — is large and soft. You fix it by smoothing.

Texture faults — a spot, a hair, a fleck of mascara — are small and sharp. You fix them by replacing.

Smooth the whole image and you destroy the pores, which is what makes retouched skin look like plastic. Heal every blotch and you get a lumpy patchwork, because the healing brush matches texture and averages tone across an area that legitimately varies.

Frequency separation splits the image into those two layers so each can be treated with its proper tool and neither affects the other.

Building it

The **low frequency** layer is a blurred copy: colour and tone with no detail. The **high frequency** layer is what the blur removed: detail with no colour or tone, held around a neutral mid-grey. Recombined, they reconstruct the original exactly.

One portrait, split so each fault meets its proper tool

high — set to Linear Light	Detail with no tone, held around neutral grey. Heal and clone here: there is no tone to get wrong, so a spot comes out without the pale disc.
low — Gaussian blurred, 5 to 10 pixels on a 20-megapixel portrait	Colour and tone with no detail. Lasso generously and blur, or paint at 10 per cent opacity. Even it out; flatten it and the face goes mask-like.
background — the original, untouched	The two layers above recombine to this exactly. The arithmetic is exact rather than approximate, which is what makes the split safe to work in.

Choose the blur radius by looking at the low layer on its own: if the spots are still visible it is too small, and if the nose has disappeared it is too large. Never work on the low layer across a hard edge — the blur has mixed the jaw and the background there, and the result is a pale halo tracing the face.

In Photoshop or Photopea:

1. Duplicate the background layer twice. Name the lower copy `low`, the upper `high`.
2. Hide `high`. On `low`, apply **Gaussian Blur**. Choose the radius by eye: large enough that the blemishes disappear, small enough that the face's structure — the nose, the eye sockets — is still recognisable. On a 20-megapixel portrait that is usually 5 to 10 pixels.
3. Show `high` and select it. **Image > Apply Image**. Set Layer to `low`, Blending to **Subtract**, Scale **2**, Offset **128**. (In a 16-bit document, use

Blending **Add**, tick **Invert**, Scale **2**, Offset **0**.)
4. Set `high` to blend mode **Linear Light**.

The image now looks identical to the original, and it is: the arithmetic is exact, not approximate.

In GIMP the classic route needs no dialogue arithmetic. Duplicate twice, blur the lower copy, and set the upper copy's mode to **Grain extract** with the blurred one beneath it; merge those two and set the result to **Grain merge** over the blurred layer. Same split, same reversibility. GIMP also ships **Wavelet Decompose**, which splits into several frequency bands at once, and the **G'MIC** plugin has *Split Details*, which is the most flexible of all.

Working on each layer

On `high`, use the Healing Brush and Clone Stamp. Spots, stray hairs, flecks. Because this layer holds no tone, healing here cannot leave a pale disc — there is no tone to get wrong. This is the single biggest practical gain of the technique.

On `low`, use the Lasso and blur, or a soft low-opacity brush. Select the blotchy region with a lasso feathered generously (20 to 40 pixels), then apply a Gaussian blur of roughly the same radius. The colour evens out and the texture above is untouched. Or sample a good colour nearby with the eye-dropper and paint at 10 per cent opacity with a very soft brush.

The discipline that keeps it looking real: on `low` you are *evening* rather than *flattening*. A face has legitimate tonal structure — the shadow beside the nose, the warmth on the cheeks. Blur that away and the face goes flat and mask-like, which is the other way this technique fails.

Where it breaks, and why

Hard edges. At the jawline, where skin meets a dark background, the blur on `low` mixes the two. The `high` layer then holds a strong edge signature in compensation. Work on `low` near that edge and you get a visible halo — a pale or dark glow tracing the jaw.

The rule that follows: **never work on the low layer across a hard edge.** Stop the lasso short of the boundary, or mask the low-layer work away from it. If you must correct right up to an edge, do it on the original rather than in the separated stack.

Never clone across an edge on the high layer either. Cloning skin texture over the boundary between face and background copies the edge signature with it, and you get a ghost of the jaw in the middle of the cheek.

Wrong radius. Too small and the blemishes survive into `low`, so you are back to fighting them. Too large and the face's structure lands in `high`, where healing it produces strange results. The test: look at `low` on its own. If you can still see the spots, the radius is too small. If the nose has disappeared, it is too large.

It does not fix shape. Frequency separation cannot move anything. A crooked nose, an awkward fold of fabric, a bulge in the wrong place is a warp problem, not a frequency problem.

Where it is actually used

Beyond skin: product packaging with printed texture, fabric with a weave, wood grain, scanned paper, painted walls, and anything where you need to correct colour without losing surface. A dented cardboard box responds very well — the dent's shading is on `low`, the print on `high`, and you can flatten the shading while keeping the printing legible.

It is also the basis of **texture transplantation**: copy a patch of good texture from the high layer of one area and paste it over a destroyed area, then correct the tone separately on low. This is how a badly damaged region of a photograph is rebuilt.

The honest caution

Frequency separation is powerful enough to make bad retouching invisible to the retoucher. The high-frequency layer preserves pores, so the result *reads* as skin even after the low layer has been smoothed into something no face does.

Two checks. Toggle the whole retouch off and on at 100 per cent: if the person has become someone else's idea of a person, reduce it. And view at the final output size, because at 400 per cent every pore is a decision and at 100 per cent almost none of them matter. Most over-retouching happens at zoom levels nobody will ever see the picture at.

BEFORE YOU MOVE ON

A retoucher evens out a red patch on a cheek by lassoing it and blurring the low-frequency layer, but the lasso extends slightly past the jawline onto the dark background. A pale glow now traces the jaw. What caused it?

A. The lasso feather was too large, so the correction bled onto the background and lightened it.

B. Blurring across a boundary mixes skin and background in the low layer, while the high layer still carries the sharp edge signature, so the recombination shows a glow.

C. Linear Light blend mode amplifies any change on the low layer near high-contrast areas, which is why the effect appears only at the jaw.

D. The Apply Image offset was set to 128 in a 16-bit document, which shifts values near the extremes and shows most at edges.

42 · 9 min

Keep what is permanent, remove what is passing

THE ONE THING TO KEEP

Remove what would be gone in a fortnight and keep what would not, do the retouch fully and then reduce the group's opacity to correct for the judgement drift that happens at high zoom, and treat eyes and teeth by removing colour rather than adding brightness.

The one rule that resolves most decisions

Remove what would be gone in a fortnight. Keep what would still be there.

A spot, a stray hair across the face, a bit of lint on a collar, a smudge of mascara, the shine from a hot light: all temporary. Nobody, including the subject, thinks of those as part of how they look.

A mole, a scar, a birthmark, freckles, laughter lines, the shape of a nose: permanent. Removing those is not retouching the photograph, it is editing the person.

The rule is not absolute — a subject may ask for a scar removed, and that is their call about their own image — but as a default it resolves ninety per cent of the judgements, and it is a defensible answer when a client asks why you did not smooth something.

An order that works

1. **Develop first.** Exposure, white balance, lens correction. Retouching on a badly white-balanced file means matching colours you are about to change.

2. **Blemishes**, on an empty layer with Sample All Layers on. Spot healing for isolated marks, clone for anything near an edge.
3. **Stray hairs.** Clone stamp with a small soft brush, sampling along the hair's own direction rather than across it.
4. **Frequency separation** for the tonal unevenness that remains.
5. **Dodge and burn** for shaping — this is where a portrait actually improves.
6. **Eyes, teeth, lips**, each locally and lightly.
7. **Global tone and colour.**
8. **Output sharpening**, at final size, masked off the skin.

Notice that the smoothing steps come before the shaping. Shaping on top of already-smoothed skin reads as drawn; shaping first and smoothing after removes the shaping you just did.

The specific areas, and what actually helps

Eyes. The whites of eyes are not white. They are a pale, slightly grey or slightly warm tone with visible blood vessels. Brightening them towards white is the single most recognisable amateur move.

What helps instead: a small **desaturation** of the redness, at 20 to 30 per cent, on a masked Hue/Saturation layer. Then a very slight lift in contrast on the iris, and a gentle burn round the outer edge of the iris, which is naturally darker and makes the eye read as round. Leave the catchlight — the reflection of the light source — exactly as it is; a painted catchlight is obvious, and removing one kills the eye.

Under-eye shadows. These are structure, not a blemish. The eye socket is genuinely recessed. Remove the shadow entirely and the face goes flat and strange. Reduce it: a healing or dodge pass at 30 to 40 per cent of what you first thought, and stop.

Teeth. The same error as eyes. Teeth are not white; they are off-white and vary. Desaturate the yellow slightly with a masked Hue/Saturation on Yellows, lift the brightness a little, and stop. Uniformly white teeth read as dentures at any size.

Skin shine. A hot spot from a flash or a window. The right tool is not a blur or a heal; it is a **burn** on the grey Soft Light layer, which brings the tone down while keeping the texture. A blurred shine leaves a smooth patch where the brightest skin texture was.

Flyaway hair. Clone along the hair's direction. For hairs crossing the background, it is usually easier to clone the background over them than to attempt to preserve them.

Fabric and background. People notice a crumpled collar and a distracting object behind the head far more than they notice a pore. Spend the time there.

The restraint mechanism: finish, then reduce

The technique that reliably prevents over-retouching, used by working retouchers: **do the retouch fully, put it on its own layer or group, then drop the group's opacity to 50 to 70 per cent.**

This works because judgement drifts during the work. After forty minutes at 200 per cent zoom every remaining mark looks intolerable, and the version that looked right at minute forty looks processed the next morning. Reducing at the end corrects for the drift without redoing anything.

The second check: look at the before and after at fit-to-screen, quickly toggled. If the change is *noticeable as a change* rather than as a better photograph, it is too much.

Consent, and who decides

A portrait is of a person, and the person has a view.

- **Ask before you start, not after.** "Is there anything you would like me to leave alone?" takes one sentence and prevents the situation where you have removed the scar somebody is proud of.
- **A commercial client is not the subject.** A brand commissioning a portrait does not own the subject's opinion about their own face. Where

those conflict, say so.

- **Children.** Retouching a child's appearance is a different matter again, and a good default is not to, beyond removing genuinely temporary marks.
- **Do not retouch to a standard.** The instruction "make them look good" does not mean "make them look like the composite face of advertising". Thinner, lighter, younger, smoother is a direction with a lot of momentum behind it, and it is worth noticing when you have drifted into it by default rather than by request.

The free path

Everything in this lesson works in GIMP and Krita. GIMP's Heal tool is good, its clone tool with *Sample merged* works on an empty layer, and Wavelet Decompose covers frequency separation. Krita's *Smart Patch* handles blemishes well and its brush engine is better than Photoshop's for dodge-and-burn work on a tablet.

On a phone, Snapseed's Healing tool and its selective adjustments cover most of what a casual portrait needs, non-destructively, free.

The limit

Retouching cannot fix a bad photograph of a person. Unflattering light, a moment caught between expressions, a lens too close to the face — those are capture problems, and every hour spent retouching them is an hour not spent taking another frame.

The fastest improvement available to almost anyone photographing people is not a retouching technique at all. It is a bigger, softer light source and a longer lens from further away.

BEFORE YOU MOVE ON

A retoucher brightens the whites of a subject's eyes and their teeth towards pure white. The result looks artificial even though the rest of the retouching is subtle. What is the underlying error?

A. The brightening was applied globally rather than through a mask, so surrounding skin has lifted with it.
B. Eyes and teeth are naturally off-white with visible variation, so raising luminance past their real range removes the variation that makes them read as tissue.
C. Brightening clips the highlight channel, and clipped highlights always read as artificial in a portrait.
D. The catchlight should have been brightened at the same time, and the mismatch between a bright sclera and a normal catchlight is what looks wrong.

43 · 9 min

The product has to be the product

THE ONE THING TO KEEP

The governing test for product work is whether the buyer receives what they saw, which permits removing sample defects, rebuilding straight edges against a path, and mirroring a symmetrical half, but never altering proportions or the colour the product actually is.

A different set of rules from portraits

With a person, the question is what is temporary. With a product, the question is simpler and stricter: **does the buyer receive what they saw?**

Everything that follows from that:

- **Dust, fingerprints, packing marks on the sample** — remove. The buyer's unit will not have them.

- **A scratch on the product itself** — leave, unless the sample is damaged and the production unit is not. Then say so in the delivery note.
- **A dent in the box** — remove if the box is a sample, keep if it is the product.
- **The colour** — this is the one that generates returns, and it gets its own section.
- **Proportions** — never. A bag that looks larger than it is arrives as a disappointment and a refund.

This is not only ethics. In most jurisdictions, showing goods that materially differ from what is supplied is a consumer protection matter, and marketplace platforms remove listings over it independently of any law.

Rebuilding a straight edge

The commonest product repair. A box edge is dinged, a bottle's shoulder has a nick, a phone's chamfer has a scratch.

Cloning freehand along a straight edge always wobbles. The method that does not:

1. Draw a **path** along where the edge should be, using the pen tool. Two anchors for a straight run.
2. Make a selection from it with a one-pixel feather, covering the damaged side.
3. Clone or heal **inside that selection**. The selection acts as a stencil, so your brushwork cannot cross the line, and the resulting edge is mathematically straight.

The same approach with a rectangular marquee handles most box work: select the region, then clone within it, and the boundary stays clean.

For a **repeating pattern** — a knurled grip, a label's repeated motif, a woven strap — do not clone. Select one clean repeat, copy it to a new layer, and move it into place. Repeat, offsetting slightly each time so the eye does not find the cycle.

Symmetry: the fastest product repair there is

Most products are symmetrical. A bottle, a jar, a shoe from the front, a piece of electronics.

Select the good half, duplicate it, flip horizontally, and slide it into place over the damaged half. Mask the join with a soft brush down the centre line. In thirty seconds you have a perfect object.

Two cautions. The lighting is usually **not** symmetrical — a key light from the left means the left side is brighter — so the mirrored half needs a tonal correction to match, typically a curve masked to it. And the label text is now mirrored, so the join has to fall outside any text.

This is legitimate: it produces an image of the product's real shape. It becomes illegitimate the moment it hides an asymmetry the product actually has.

Labels, and getting them flat

A label on a curved bottle is distorted by the curve and often by the camera angle. Two fixes:

Perspective warp on the whole product to correct a camera-angle error. Photoshop's *Edit > Perspective Warp*, GIMP's *Tools > Transform > Perspective*.

Replacing the label entirely, which is what is usually done for a catalogue. Take the flat artwork from the client, place it as a smart object, and use *Edit > Transform > Warp* with a cylindrical preset to bend it round the bottle. Set it to Multiply or use a luminosity-based mask so the bottle's real shading shows through. The result is a legible label with the real product's light on it.

That second technique also solves the common problem of a client whose only photograph was shot before the packaging was redesigned.

Colour, and why it is the expensive one

The returns line on a shop's accounts says "colour not as pictured" more than anything else.

The method:

1. Put a **colour checker** in the first frame of each setup, and white balance from it.
2. **Hold the real product next to the screen in daylight** and compare. This is the only test that matters, and it takes ten seconds.
3. Correct with a **Hue/Saturation layer targeted at the relevant colour band**, masked to the product, rather than a global move.
4. **Soft proof** if the image is going to print.
5. Check on a phone, because that is where it will be bought.

The one to be wary of is a saturation boost applied to make a product look appealing. A teal that is two shades brighter than the real fabric is the definition of the problem.

Background, shadow and the set

- **White** for marketplaces that require it, at exactly 255 across the background; several platforms check automatically and reject files that are not.
- **Around 248 to 250** on your own site, so the product's own white highlights read as highlights rather than dissolving.
- **Keep the contact shadow.** An object with a clean edge and nothing beneath it floats, and everybody sees it without knowing why. If the real shadow is unusable, build one: a soft ellipse on a layer set to Multiply at 20 to 25 per cent, slightly narrower than the base of the object, darker where the object meets the surface and fading outwards.
- **Same treatment across the set.** Shadows on all or on none.

The free path, and the honest limit

Everything here works in GIMP, Krita and Photopea. The pen tool, clone stamp, transform tools, blend modes and masks are all present. Photopea additionally handles PSD mockup templates, which is where most e-commerce work starts.

The limit is upstream of all of it. A product photographed under one bare bulb against a crumpled sheet cannot be retouched into a catalogue image, because the light is wrong and light is what makes a product look like an object rather than a cut-out. Ten minutes rearranging a window, a white card and a sheet of paper improves the result more than two hours of retouching, and that is the trade to make every time.

BEFORE YOU MOVE ON

A retoucher mirrors the clean left half of a symmetrical bottle over its damaged right half. The shape is now perfect, but the bottle looks subtly wrong. What is the most likely cause?

A. Mirroring reverses the pixel order, which inverts the direction of any noise grain and reads as a mismatch.

B. The key light came from one side, so the mirrored half now carries the wrong brightness and the shading reads as coming from two directions at once.

C. The mask down the centre line was too soft, so the join is visible as a band of doubled detail.

D. Mirroring a product is a misrepresentation, so the image is technically fine and the discomfort is about honesty rather than appearance.

A reflection is a photograph of the light source

THE ONE THING TO KEEP

A specular reflection is an image of the light source added on top of the surface, so it can be reduced by subtractive adjustments or replaced from a second frame shot with the reflection moved, but removing a dense one means inventing what the recorded pixels no longer contain.

What you are looking at

A specular reflection is not a property of the object. It is an image of whatever is in front of it — the window, the softbox, the room, you and your camera.

That single fact changes the approach. A reflection cannot be "cleaned off" a surface, because underneath it there is no clean version; the recorded pixel is a sum of the surface and the reflection. Removing it means deciding what should have been there and putting that in.

Three things follow:

- The easiest fix is always at capture, by changing what is being reflected.
- Reducing a reflection is far more achievable than removing it.
- Removing a large reflection is reconstruction work, with all the honesty questions that carries.

Fixing it at capture

Move the light, not the object. A reflection appears where the angle of incidence equals the angle of reflection. Move the light source thirty degrees and the hot spot moves off the product entirely. This is the whole of product lighting in one sentence.

Make the source larger and further away. A big soft source produces a broad, gentle gradient across a glossy surface — which reads as shape — instead of a small hard blob, which reads as a blemish. A window with a net curtain is a large soft source and costs nothing.

A polarising filter. Rotating a circular polariser on the lens cuts reflections from non-metallic surfaces — glass, water, plastic, painted wood, foliage — by a substantial amount, and does nothing to reflections from bare metal, because the physics differ. It costs one to two stops of light. For photographing a framed picture, a shop window or water, it is the correct tool and no editing matches it.

For spectacles, the professional answer is two frames: one normal, one with the lenses removed from the frames or with the arms lifted so the frames sit higher and the lens reflections move. Composite the eye region from the second frame. Attempting to paint out a reflection across an eye by hand is an hour and looks it.

For a screen — a phone or monitor in the shot — shoot the screen off and composite the screen content in afterwards from the actual artwork. Every clean device photograph you have seen was made this way.

Reducing a reflection in the editor

When you have one frame and a reflection that is not too dense.

The blend-mode approach. A specular reflection adds light. So work on it with tools that subtract: a Curves layer pulling down, masked precisely to the reflection, is often enough. If the reflection has colour — a blue window reflected on a red bag — a Hue/Saturation layer masked to it, reducing saturation and shifting hue, restores the surface colour underneath.

Frequency separation. A soft reflection is almost entirely low frequency. Even it out on the low layer and the surface texture above survives intact. This is the best tool for reflected sheen on leather, fabric and painted metal.

Clone from a matching area. On a surface with a consistent finish, another part of the same surface at the same angle gives you correct pixels. Sample carefully; a surface curving away has different shading, and cloning from it produces a visible patch.

Rebuilding the pattern. Where the reflection sits over a printed pattern, take the pattern from elsewhere, warp it to fit, and reapply the shading with a burn layer.

Where removal becomes reconstruction

A dense reflection — a white window across half a black television screen, a whole room reflected in a kettle — has obliterated the information underneath. Every channel in that area is near maximum. There is nothing to recover.

At that point you are not retouching; you are drawing what you believe was there. That is legitimate for a product's own glossy surface, where you know what it looks like. It is not legitimate when the reflection covers something whose content matters — a document under glass, a face behind a window, the contents of a display case.

The test from earlier in the course applies: would the viewer be surprised to learn what you did? Rebuilding a kettle's steel finish, no. Inventing what was inside a shop window, yes.

The two-frame method, generalised

The most useful technique in this lesson, and it costs nothing.

Put the camera on a tripod or a stack of books. Take one frame. Change what is being reflected — move the lamp, hold a black card between the light and the object, drape a dark cloth behind the camera. Take a second frame.

Now the reflection is in a different place in each frame, and the correct pixels for one are in the other. Load both as layers, align them (the Difference blend mode makes misalignment visible), and mask between them.

The extreme version of this is **black cloth flagging**: photographing a glossy product with a large piece of black fabric behind the camera, so the object reflects darkness rather than the room. Two metres of black cotton is a few pounds and solves more product photography problems than any filter.

Free tools, and the limit

Nothing here needs paid software. GIMP, Krita and Photopea all have the clone, heal, curves, masks and blend modes required. The polariser and the black cloth are hardware and cheap.

The honest limit: there is research into single-image reflection removal using neural networks, and there are apps that claim it. On a photograph through a window it sometimes produces a plausible result, and plausible is the operative word — the network is inferring what was behind the glass, not measuring it. For a holiday snapshot through an aeroplane window, fine. For anything where the content behind the glass is the point, what you get back is a guess presented as a photograph, and the earlier lessons on upscaling and denoising apply in exactly the same way.

BEFORE YOU MOVE ON

A photographer uses a circular polariser to cut reflections when photographing a display of jewellery in a glass case. The reflections on the glass reduce substantially, but the hot spots on the polished silver do not change at all. Why?

A. The polariser was rotated to the optimal angle for the glass, and metal requires the opposite rotation.

B. Reflections from metal surfaces are not polarised in the way reflections from dielectrics are, so a polariser has no reflection to remove there.

C. The silver is inside the case, so the polariser only acts on the outermost reflecting surface in the optical path.

D. Specular highlights on metal are brighter than the filter's one to two stops of loss can suppress, so they survive the exposure reduction.

45 · 9 min

Rebuild the structure first, then the texture

THE ONE THING TO KEEP

Patch-based fill copies texture from elsewhere in the same photograph and has no model of geometry, so large removals are done by rebuilding the structural lines by hand first, letting the fill handle only flat texture, and then unifying tone and grain across the repair.

Why big removals fail

Small removals succeed because the surroundings are uniform. A spot on a wall is surrounded by wall.

A large removal — a person in front of a doorway, a car on a street, a pylon across a horizon — fails because the surroundings are *structured*. Content-aware fill smears, generative fill invents, and cloning leaves a repeated patch, all for the same reason: none of them knows what the building's geometry was.

The method that works reverses the usual order. **Rebuild the structure first, by hand, then borrow texture to cover it.**

What content-aware fill is actually doing

It is worth knowing so you can predict it. The underlying algorithm in most implementations is a patch-based synthesis — the best-known is **PatchMatch**, published in 2009. For every small patch in the hole, it searches the rest of the image for the most similar patch and copies it in, then iterates so neighbouring patches agree with each other.

Everything it puts in was somewhere else in your photograph. It has no model of the world, only of your pixels.

So the failures are predictable:

- **Unique content cannot be reconstructed.** Remove a person standing in front of a doorway and it fills with wall, because there is no other doorway to copy.
- **Straight lines break.** A window frame, a skirting board, a horizon continues only if a similarly-angled fragment exists nearby, and the patch search does not understand "continue this line".
- **Perspective is ignored.** It copies a patch of paving from the foreground into the middle distance at the wrong scale.
- **Repeating textures come back visibly repeated**, because the same good patch is the best match many times over.

The controls that help: Photoshop's *Content-Aware Fill* workspace lets you paint which areas it may sample from — exclude the foreground when filling a distant area. GIMP's **Resynthesizer** plugin, which predates Adobe's feature and is free, has similar controls and is genuinely good.

The manual method, in order

Take a photograph of a street with a parked van you need gone.

1. Reconstruct the large geometry. Identify the lines that must continue through the hole — the kerb, the wall base, the roof line, the far pavement edge. For each, find a clean section elsewhere in the frame, copy it onto a new layer, and move it into position. Use *Edit > Free Transform* with perspective to match the angle. You are not worrying about texture yet; you are re-establishing where things are.

2. Use the vanishing point or perspective clone for anything receding. Photoshop has *Filter > Vanishing Point*, where you define a perspective plane and then clone within it, and the tool scales and skews the copied pixels to match. GIMP has the **Perspective Clone** tool, which does exactly the same job and is one of GIMP's genuinely strong features. Without either, transform a copied patch manually.

3. Fill the remaining flat areas with content-aware fill or Resynthesizer, now that the structure is in place. These tools are good at texture and bad at geometry, so give them only the texture job.

4. Unify the tone. The borrowed pieces came from different parts of the frame with different brightness. A masked Curves layer over the repaired region, matched against its surroundings, removes the patchwork look. This step is skipped more than any other and it is what makes a repair invisible.

5. Add back grain and noise. Cloned and synthesised areas are cleaner than the photograph around them. A noise layer at 1 to 3 per cent, masked to the repaired region and matched to the image's own grain, is the final tell removed. The compositing block covers matching grain properly.

6. Check at 100 per cent and at fit-to-screen. Repairs that look fine close up often show as a differently-toned block at small size, and repairs that look fine small often have obvious repeats close up.

Generative fill: where it fits

A diffusion model asked to fill a masked region will produce plausible new content rather than borrowed content. For a removal it is often the fastest route, and it comes with the constraint stated across this course: it is inventing.

Two practical notes beyond the honesty question.

It works best on organic, low-stakes content — foliage, sky, gravel, water, crowd. It works worst on anything with regular geometry or text, where it produces convincing-looking nonsense: a shopfront with letters that are not letters, brickwork with courses that do not line up.

Resolution. Most generative fill implementations work on a limited region at a limited resolution and then scale the result in. On a large file the filled area can be softer than its surroundings, which is a giveaway. Filling in several smaller passes rather than one large one helps.

What determines whether you should

The judgement from the earlier retouching lesson, applied to size: **does removing this change what the picture asserts?**

A bin in the corner of a wedding photograph, no. A brand's logo on a competitor's product in an editorial photograph, yes. A person from a news photograph, absolutely — that has ended careers, and it is the clearest line in the trade.

For commercial work the practical safeguard is a note in the delivery: "vehicle removed from left of frame, background reconstructed." One line, and nobody is surprised later.

The limit

Some removals are not possible, and recognising them early is worth more than technique. The test: **is what was behind the object visible anywhere else in the frame, or reconstructible from geometry?**

If a van covers a shop's entire frontage and no other frame shows it, the answer is no. What can be produced is an invention that looks like a shop. Whether that is acceptable depends entirely on what the picture is for, and the person to ask is the client, before you spend the afternoon.

BEFORE YOU MOVE ON

An editor selects a person standing in front of a shop doorway and runs content-aware fill. The doorway is replaced with more wall. What does this tell you about the tool?

A. The sampling region was set too narrowly, and widening it to the whole image would let it find the doorway.
B. The fill operates at a reduced resolution and reconstructs only low-frequency content, which is why fine architectural detail is lost.
C. It synthesises the hole from patches found elsewhere in the same photograph, and since no other doorway exists in the frame, wall is the best available match.
D. Content-aware fill prefers uniform regions by design, so it will always choose the simplest plausible fill even when more detailed material is available.

46 · 9 min

Moving pixels, and the straight line that gives it away

THE ONE THING TO KEEP

A warp is a displacement mesh that moves everything within its influence, so the reliable evidence of reshaping is a bent straight line in the background — which is exactly what the freeze mask, or warping the subject on its own layer, exists to prevent.

What a warp actually is

Liquify, Puppet Warp, Free Transform's Warp and Perspective Warp all do one thing: they build a **displacement map**, a grid saying where each pixel should move to, and then resample the image through it.

No pixels are created and none are destroyed; they are relocated and interpolated between. That is why a warp softens whatever it stretches — the stretched region is interpolated across more pixels than it originally occupied — and why an area compressed by a warp becomes slightly sharper and can alias.

The tools, and what each is for

Liquify (*Filter* > *Liquify*) gives you brushes that push the mesh around.

- **Forward Warp** pushes pixels in the direction you drag. The main tool.
- **Pucker** and **Bloat** contract and expand around the brush centre.
- **Freeze Mask** paints a region the warp cannot touch. This is the control that makes Liquify usable — freeze the background before you move anything near it, and the straight lines behind your subject survive.
- **Reconstruct** brings a region back towards the original.

Apply Liquify to a **smart object** and it becomes a re-editable smart filter: re-open it a week later with your mesh intact. GIMP's equivalent is the **Warp Transform** tool (*Tools > Transform > Warp Transform*), which has the same brushes including a Freeze-like erase mode. Krita has a Liquify mode inside its Transform tool. Photopea has Liquify.

Puppet Warp places pins and lets you move them, with the mesh deforming smoothly between. Correct for bending an arm, straightening a drooping flower, adjusting the curve of a strap. Pin the parts that should not move first; most Puppet Warp disasters are from too few pins.

Perspective Warp lets you define quads on planes in the image and then adjust them, which is how you change the apparent camera angle on a box or straighten a building face.

Free Transform > Warp gives a simple grid, good for wrapping a label round a bottle or fitting artwork to a curved surface.

The uses nobody argues about

- **Straightening a label** that was applied crooked on the sample.
- **Fixing the stretch at the frame edge.** A wide lens stretches anything near the corner — a face photographed at the edge of a group shot is genuinely wider than it is. Correcting that restores the person's real shape rather than altering it.
- **Adjusting a fold of fabric** so a garment hangs as it would on a person.
- **Closing a gap** where a badly fitted lid or a twisted strap shows on a sample.
- **Matching two composited elements** whose perspectives disagree slightly.
- **Straightening a horizon that bends** after a heavy distortion correction.

The tell: warping bends the background

The technical point that makes this lesson practical. A displacement mesh affects everything inside its influence, including whatever is behind the subject.

So a body reshaped against a tiled wall leaves the tile grout curving. A waist narrowed in front of a door frame leaves the frame bowed. A head slimmed against a window leaves the mullion bent. This is how retouching is spotted, every time, and it is why the Freeze Mask exists.

The practice that avoids it:

1. **Freeze the background** before you begin, or duplicate the subject onto its own layer and warp only that.
2. After warping, **look only at the straight lines** in the frame at 100 per cent. Walk every edge, every horizon, every architectural line.
3. Toggle the layer off and on. Any line that moves is a line you have bent.

It is also how you check somebody else's work, which comes up more than you might expect.

The line, restated for shape

Earlier in the course the test was: would the viewer be surprised to learn what you did? Applied to shape:

- **Correcting a distortion the lens introduced** restores what was there. Nobody is misled.
- **Changing what the subject's body is shaped like** asserts something about a person that is not true. Several countries require commercial images to be labelled where body shape has been altered, and the direction of legislation has been one way for over a decade.
- **Changing a product's proportions** is the consumer protection issue from the product lesson.
- **Changing a scene's geometry** — moving a building, straightening a road — in a documentary photograph is the same category as removing an object from one.

The practical rule for commercial portrait and fashion work: **ask, and write down the answer.** "Do you want any reshaping?" is one question. If the answer is no, you have a record. If the answer is yes, you have consent, which is what distinguishes a service from an imposition.

For your own work, the honest observation is that liquify is habit-forming. It is easy, invisible while you are doing it, and there is no natural stopping point. Working on a duplicate layer whose opacity you reduce at the end, exactly as with retouching, is the most effective brake anybody has found.

The technical limits

Warping softens. Heavy stretching interpolates, and the stretched region will not match the sharpness of its surroundings. Sharpen locally afterwards or keep the warp small.

Warping repeats noise. Stretched grain becomes elongated grain, which is visible against unstretched grain nearby. On a noisy file this is a real tell.

Warping cannot fix occlusion. If an arm is in front of a bag, no warp reveals the bag. That is a reconstruction job.

A mesh has a resolution. Liquify's mesh is coarser than the pixels, so very fine, very local moves come out smeared rather than precise. For a small precise move, a selection, a nudge and a masked blend is cleaner than a warp.

BEFORE YOU MOVE ON

A retoucher narrows a subject's waist with Liquify's Forward Warp brush. The body looks convincing, but a reviewer immediately identifies the edit. What did they almost certainly notice?

A. The warped area is softer than the surrounding skin, because stretching interpolates across more pixels than the original occupied.
B. The vertical lines of the wall or door frame behind the subject are curved, because the displacement mesh moved the background along with the body.
C. The grain in the warped region runs in a different direction, which reads as a patch against the surrounding noise.
D. The subject's proportions no longer match a plausible human figure, which is the standard way such edits are detected.

47 · 9 min

Getting the photograph off the paper

THE ONE THING TO KEEP

Scan resolution is chosen from the intended output size with room for correction, moiré from a previously printed original is an interference pattern best removed by oversampling and downscaling, and infrared dust removal fails on silver black-and-white and Kodachrome because the image itself blocks infrared.

Resolution: scan for the output, with room

A flatbed scanner's dpi setting is real, unlike a JPEG's ppi tag — it genuinely determines how many samples are taken per inch of the original.

The arithmetic you need:

- **A 6×4 inch print reprinted at the same size** needs 300 dpi minimum. Scan at **600** so you have room to crop, straighten and correct.

- **A 6×4 print to be enlarged to A4** — that is roughly a 2.5× enlargement, so scan at **800 to 1200 dpi**.
- **A 35 mm negative or slide** is about one inch across, so a usable file needs **2400 to 4000 dpi**, and that requires a scanner with a transparency unit. A flatbed's stated 9600 dpi is usually interpolated marketing rather than real optical resolution; look for the *optical* figure.
- **Above the scanner's optical resolution there is no gain**, only file size. The sensor cannot sample more finely than it can sample.

Scan in **16-bit** if the option exists. An old print is often faded and low in contrast, which means the correction will be a heavy stretch, which is exactly the case for extra bit depth.

Save the raw scan as a **TIFF** and never overwrite it. That file is the closest thing to the original that will exist once the print fades further.

Moiré, and the print that was already printed

Scanning a photographic print from a lab is straightforward. Scanning something from a **magazine, newspaper or book** is not, because that image is made of a regular grid of halftone dots.

When the scanner's sampling grid meets the printing screen's grid, the two interfere and you get **moiré** — coloured wave patterns across the image. This is the same aliasing mechanism as the demosaic moiré from the first module, and it is not a fault.

Three fixes, in order of quality:

1. **Scan at a much higher resolution than you need, then downsample.** Scanning at 1200 dpi and reducing to 300 averages many samples per printed dot, which removes the interference properly. This is the best result and the largest files.
2. **Use the scanner software's descreen setting**, naming the original type (newspaper, magazine, book). It applies a targeted blur. Fast, and it costs sharpness.

3. **Rotate the original a few degrees on the glass**, scan, and rotate back in software. Changing the angle between the two grids breaks the interference pattern. Crude, effective, free.

In the editor, a small Gaussian blur followed by sharpening also suppresses residual moiré, and a **Median** filter at 1 to 2 pixels is often better than a blur because it removes the dot pattern while keeping edges.

Dust and scratches

Clean the glass and the print before anything else. A microfibre cloth and, for the print, a soft brush. Two minutes here saves an hour of healing.

Infrared dust removal — sold as Digital ICE, FARE, or similar — uses a fourth infrared channel to detect physical defects, because dust and scratches block infrared while the image dyes do not. It is excellent and it has two exceptions that catch everybody: it does **not** work on traditional silver black-and-white negatives, or on Kodachrome, because in both the image itself is opaque to infrared and gets detected as damage. Turning it on for those produces a smeared mess.

For prints, infrared removal generally does not apply — it is a film-scanning feature. Prints are cleaned by hand in the editor.

Flatness, and photographing instead

A curled print will not sit on the glass and scans with a soft band. Put it under a heavy book overnight, or scan with the lid pressed down and a weight on top.

No scanner? A phone works, and the technique matters:

- **Even light, no direct source.** Two lamps at 45 degrees from opposite sides, or an overcast window, or outdoors in open shade. A single overhead light produces a glare patch.
- **Camera square to the print.** Tilt introduces keystone which then has to be corrected, and correcting it resamples.

- **Avoid the phone's own shadow**, which is why the light comes from the sides.
- **Use the main camera, not the ultrawide**, whose distortion bends the print's edges.
- **Shoot DNG if you can**, for the bit depth on a faded original.
- Crop and correct perspective afterwards. Most phone galleries and Adobe Scan do this automatically, and doing it manually with a perspective transform gives a better result.

A phone photograph in good light beats a bad scan, and both beat a photograph taken at an angle under a ceiling light.

Free software for scanning

XSane and **Simple Scan** on Linux, **NAPS2** on Windows, and **VueScan** across platforms (paid, and it supports very old scanners whose manufacturers abandoned drivers years ago, which is often the only thing that will).

GIMP can drive a scanner directly through XSane. For negatives, **darktable** has a *negadoctor* module that inverts and corrects colour negative film scans properly, accounting for the orange mask — this is a genuinely hard job done well, and free.

What the scan cannot recover

A scan captures what is on the paper now. If the print has faded, the scan captures the faded version; if it has yellowed, the scan captures the yellow.

Some of that is correctable, which is the next lesson. Some is not: where a dye has faded to nothing, there is no residual signal to amplify, and lifting that area produces noise and paper texture rather than a picture. A print left in a window for thirty years may have a corner with genuinely no image left.

And resolution cannot exceed what the print holds. A small print from a 1970s compact camera on cheap paper may contain very little detail; scanning it at 2400 dpi produces a large file of magnified paper grain. Scan it at 600, and spend the time on the tonal restoration instead.

BEFORE YOU MOVE ON

An archivist scans a photograph from an old magazine at exactly 300 dpi for a 300 dpi reprint, and the result shows coloured wave patterns across the image. What is the best fix?

A. Enable infrared dust removal, which detects the printing screen as a surface defect and removes it.
B. Scan at 1200 dpi and downsample to 300, so many samples average across each printed dot and the interference disappears.
C. Scan at 150 dpi to sample below the printing screen's frequency, then upscale to 300.
D. Increase the scanner's colour depth to 16-bit, which gives enough levels to represent the dot pattern smoothly.

48 · 9 min

Tears, mould and the cyan that faded first

THE ONE THING TO KEEP

Restoration runs large damage to small, the pink cast on old colour prints is the cyan dye layer having faded and can be neutralised but not recovered, and anything reconstructed rather than derived from surviving evidence belongs in a written note delivered with the file.

An order that keeps you out of trouble

Restoration goes wrong when people start with the interesting part. The order:

1. **Scan, and archive the raw scan untouched.** Everything after this is a copy.
2. **Global tone.** Recover contrast from a faded original before judging anything else — damage that looked catastrophic often turns out to be faint.

3. **Global colour.** Correct the cast. Again, before the detail work, because you will be matching colours.
4. **Large structural damage** — tears, missing corners, holes. Reconstruct geometry first, exactly as in the object-removal lesson.
5. **Medium damage** — creases, stains, mould blooms, silvering.
6. **Small damage** — dust, scratches, spots.
7. **Local tone**, to even out what the repairs left.
8. **Grain matching**, so the repairs sit in the original's texture.
9. **Output**, sized and sharpened for whatever the family actually wants.

Working small-to-large means redoing the small work after every large repair. Working large-to-small means you only do it once.

Why old colour prints go red

A useful piece of mechanism. Colour prints and negatives hold three dye layers — cyan, magenta and yellow. They do not fade at the same rate. In the common chromogenic processes the **cyan dye is the least stable**, so it goes first.

Remove cyan and what remains is magenta and yellow, which together read as red-orange. That is why every 1970s and 1980s family photograph has the same salmon-pink cast. It is not the paper yellowing; it is a specific dye layer failing.

The correction follows directly: you need to **add back cyan**, which means reducing red. A Curves layer with the Red channel pulled down in the midtones, or a Colour Balance shifted towards cyan, targeted at the tonal range where the cast is worst — usually the midtones and highlights rather than the shadows.

The limit is equally direct. Where the cyan dye has faded completely, there is no cyan information to restore. You can neutralise the cast, and what you get back is a grey-ish, low-saturation image rather than the original colours, because the data in that channel is gone. Pushing further produces noise and channel banding, not colour.

For badly faded prints, converting to black and white is often the honest and better-looking answer. Say so to the family; most prefer a good monochrome to a bad approximation of colour.

The specific damage types

Tears and missing corners. Reconstruct the geometry by hand. Where a corner is gone entirely and contained nothing but background, clone from the opposite corner. Where it contained content, you are inventing, and that needs to be said.

Creases. A crease is a tonal line, not missing content — the emulsion is cracked but the image either side survives. The Healing Brush along the crease, in short strokes following its direction, works well. Where it crosses a face, work on a frequency-separated pair so you fix the tone on low and the texture on high.

Mould and foxing — brown spots, usually round. These are tonal and colour damage over intact image. A masked Curves plus a desaturation, or a patch from a clean area with the tone corrected, both work. Heavy mould has eaten the emulsion and is a reconstruction.

Silvering — the mirror-like metallic sheen on the dark areas of old black-and-white prints, caused by silver migrating to the surface. It appears as a blue-grey iridescence. Scan with the print slightly angled to reduce it, then desaturate the affected areas and rebuild the tone. It cannot be fully removed because the surface is physically altered.

Water damage and staining. Usually a large soft tonal and colour shift. A masked Curves layer with an inverse gradient often removes most of it in one move, which is far better than cloning.

Fading at the edges from a frame or an album produces a rectangle of different density. Again tonal, again a masked adjustment rather than cloning.

Rebuilding what is missing

Three sources, in order of honesty:

Symmetry. A missing half of a shirt collar, a chair arm, a window frame: mirror the good side. Real content, correctly derived.

Elsewhere in the same photograph. A missing patch of grass, wall, sky, fabric. Borrowed content from the same moment.

Another photograph of the same subject. A missing ear from another frame of the same person on the same day. This is what professional restorers do, and it is why the first question to a family should be "are there other pictures from that day?"

And then the fourth, which is different in kind: **inventing it**, by hand or with a generative model. That produces a plausible ear that is not their ear. For a family photograph on a wall, many people are content with that and they should be told it is what happened. For anything archival, historical or evidential, it is not acceptable, because it introduces fiction into a record.

Documenting what you did

The practice that separates restoration from alteration, and it costs five minutes:

- Keep the untouched scan, named clearly, and deliver it alongside the restoration.
- Write a short note: "Tear across lower left reconstructed from surrounding wall. Missing corner filled with background. Colour cast corrected; original cyan layer substantially faded, so colours are approximate. Child's face on right rebuilt partly from the second photograph supplied."
- Deliver the layered file if the client can use it.

Archives and museums require this. Families appreciate it. And it protects you, because in ten years somebody will ask whether great-aunt's brooch was really that shape.

The free path

GIMP with Resynthesizer covers structural fill. G'MIC has a large set of restoration filters including inpainting and dust removal. darktable's negadoctor handles film negatives. Krita's Smart Patch is good for spot work. Everything in this lesson is achievable at no cost, and restoration is an area where patience matters far more than the software.

What cannot be restored

Where the emulsion is gone, there is no image. Where a dye has fully faded, there is no colour. Where a print was out of focus, it is out of focus. Restoration recovers what survives and reconstructs what can be derived. Everything beyond that is drawing, and drawing on a photograph is a different act that deserves a different name and a sentence in the delivery note.

BEFORE YOU MOVE ON

A 1978 family print has a strong salmon-pink cast. After correcting it with a Curves move on the red channel, the image is neutral but the colours are weak and desaturated, and pushing saturation produces noise. Why?

A. The scan was 8-bit, so the correction exhausted the available levels; rescanning in 16-bit would restore the colour.
B. The cyan dye layer has substantially faded, so neutralising the cast removes the imbalance without restoring information that is no longer in the print.
C. The correction should have been applied to the cyan channel rather than the red channel, which is why the result is weak.
D. Curves applied to a single channel always reduces saturation, and a Colour Balance adjustment would have preserved it.

49 · 9 min

Four losses that no tool undoes

THE ONE THING TO KEEP

Blur, defocus, clipping and compression are absences of information rather than faults in it, so every tool that appears to reverse them is generating plausible content — and a five-minute triage at 100 per cent zoom tells you on day one what a file can become.

Knowing this is a professional skill

The most expensive mistake in this trade is spending four hours on a file that could not be fixed, and telling the client on Friday. Recognising an unrecoverable file in the first five minutes is worth more than any technique in this course.

There are four main losses, and each has a mechanism that explains why it is permanent.

Absences of information, and what the button that claims to fix them does

	Why it is permanent	What an enhance button does
blur and defocus	**The kernel is** … d deconvolution is ill-p…	**Draws edges nobody measured** letters come back as the wrong letters, faces as a different person
…avy compression	**Every pixel holds** … …g distinguishes one pixel from the …	**Invents texture over the flat patch** plausible, and not a measurement of anything

These are not faults in the data; they are places where there is no data. Every tool that appears to reverse them is generating content consistent with what survives, and it looks equally confident whether the evidence was strong or nearly absent. Five minutes at 100 per cent on day one is worth more than any technique.

1. Motion blur and camera shake

A blurred photograph is the sharp image **convolved** with a blur kernel — the path the camera or subject travelled during the exposure. Undoing it is deconvolution.

If you know the kernel exactly, deconvolution works mathematically. Astronomers and microscopists do it routinely because they can measure their optical system's point spread function precisely.

For an ordinary photograph you do not know the kernel, so the problem is **blind deconvolution**: estimating both the kernel and the sharp image from the blurred result alone. That is ill-posed — many combinations of kernel and image produce the same blur — and in the presence of noise the estimate amplifies the noise enormously.

What this means in practice:

- **Simple, short, linear motion blur** can be partially undone. Photoshop's *Shake Reduction* and GIMP's G'MIC deconvolution filters do genuinely recover some sharpness on mild cases.
- **Anything more** — long exposure, rotational shake, a subject moving while the camera moved, multiple objects at different speeds — produces ringing artefacts worse than the blur.
- **Model-based deblurring** produces a sharp-looking image by generating plausible detail. It is not recovery; it is the same mechanism as AI upscaling, with the same consequences for anything evidential.

2. Out of focus

Same mathematics, different kernel — a disc rather than a streak — and worse, because defocus blur discards information more thoroughly. Fine detail below the blur's scale has been averaged to a uniform value, and an average does not retain what was averaged.

An additional problem: focus varies with depth. A photograph focused on the background has different blur on the foreground, so there is no single kernel for the frame.

Sharpening does not help, because sharpening exaggerates edges that exist and a defocused image's edges are gone. Applying heavy sharpening to a soft photograph gives you a soft photograph with halos.

The practical answer is always the same: if the eyes are not sharp, the frame is not usable for anything that will be seen large. It may still be fine at 400 pixels on a web page, and that is a legitimate use.

3. Clipped highlights and crushed blacks

When all three channels reach 255, every pixel in that region holds the same value. A blown sky is a uniform patch of 255 with no variation — there is nothing to recover because nothing distinguishes one pixel from another.

Raw gives you the partial case: where one or two channels clipped and one did not, the missing ones can be reconstructed. That is the genuine one-to-two-stop headroom from the raw lesson. Where all three are at maximum, no format helps.

What can be done is **replacement**: putting a sky from another frame in, or painting a gradient. That is compositing, it should look deliberate, and it should be disclosed if the picture is documentary.

Crushed blacks are the same at the other end, and they are usually less damaging because the eye expects shadows to contain little.

4. Compression and resolution

A heavily compressed JPEG has had its transform coefficients rounded to zero. A small image never sampled the detail. Both are absences, and both attract tools that claim to restore them.

Deblocking filters smooth the artefacts, which trades one loss for another. Model-based restoration generates plausible detail from the low-quality evidence. As established in the first module, that is invention that looks like recovery, and the failure modes are specific: letterforms come back as the wrong letters, faces come back as slightly different people, repeated textures come back as convincing patterns that are not the pattern that was there.

For a family photograph destined for a frame, plenty of people are happy with the result and should be told what it is. For a document, a number plate, a medical image, a legal exhibit or a news photograph, it is not usable, because what comes out is a guess wearing the appearance of a fact.

The five-minute triage

Before accepting a repair job, check in this order:

1. **Open at 100 per cent and look at the most important detail** — usually the eyes, the label, the face. Sharp or not sharp? This is the fastest single test.
2. **Histogram, three-channel.** Are the highlights piled at 255 across all three? Are the shadows at 0?
3. **Zoom to 200 per cent on a flat area.** JPEG blocks? Heavy noise? Both cap what the file can become.
4. **Pixel dimensions against the intended output.** Divide the pixel width by the printed inches. Under 120 and you have a conversation to have.
5. **Ask whether an original exists.** Most "unusable" files are WhatsApp copies of perfectly good originals, and one message solves it.

That is five minutes and it converts a bad week into a clear answer on day one.

How to say it

"This can't be fixed" is rarely the most useful sentence. Better:

> *"The eyes are out of focus, so it won't hold up printed larger than about 10 by 15 centimetres. I can correct the colour and the damage, and it will look good at that size. If you need it bigger, is there another frame from the same day?"*

That names the limit, states what is possible, gives a size, and asks the question that most often solves the problem. It also protects you, because the expectation is now written down.

The honest general statement to carry out of this module: **editing redistributes and repairs information; it does not create it.** Everything that appears to break that rule is a model inventing plausible content, and the only question worth asking about it is whether the picture is allowed to contain invention. That question is answered by what the picture is for, not by the tool.

BEFORE YOU MOVE ON

A client brings a photograph with visible camera shake and asks for it to be sharpened. Why is deconvolution able to help an astronomer with a comparable blur but not this client?

A. Astronomical images are monochrome, and deconvolution cannot be applied to colour data without introducing channel artefacts.

B. Astronomical sensors capture far more bits per pixel, so there is enough precision to invert the blur that an ordinary camera lacks.

C. An astronomer can measure the optical system's point spread function, so the blur kernel is known, while an unknown shake path makes the inversion ill-posed and noise-amplifying.

D. Astronomical exposures are long and steady, so their blur is uniform across the frame while camera shake varies by region.

CASE STUDY · MODULE 5

The 1971 wedding photograph, and the face that is only in one other frame

A small photo studio near Naveen Market in Kanpur, where restoration pays better per hour than passport photographs.

What came in

A colour print, five inches by seven, from a wedding in 1971. Salmon-pink overall. A tear running up from the lower left corner, and the corner itself missing — about two centimetres of what had been a plain wall. And a crease, folded and refolded, running diagonally across the grandmother's left eye and cheek.

The pink is not the paper yellowing. Chromogenic colour prints hold three dye layers, and they do not fade at the same rate; in the common processes the cyan goes first. Remove cyan from cyan, magenta and yellow and what is left reads as red-orange. That is why every family photograph from this period has the same cast, and it is why the correction is a red-channel curve pulled down in the midtones rather than a warmth slider.

The limit is equally specific. Where the cyan dye has gone, there is no cyan information to amplify. You can neutralise the cast and what you get back is a low-saturation, greyish image, because the data in that channel is absent rather than weak. Pushing further produces channel noise and banding, not colour.

The tear and the missing corner were straightforward: a tear is a tonal line over surviving image, and the corner had contained nothing but wall, so it could be cloned from the opposite corner. Real content, correctly derived.

The crease across the eye was the job.

The decision

The family had brought a second print from the same day. It is badly out of focus and it shows the grandmother from the other side, three-quarters rather than near-frontal.

Rebuild the eye from the second print. Six hours: extract the eye region, warp it to the new angle, match the grain and the tone, sharpen it to sit with its surroundings. The result would be derived from surviving evidence of that person on that day. Six hours is not in a two-and-a-half-thousand-rupee restoration.

Generate or hand-paint an eye. Forty minutes. It would be a plausible eye. It would not be her eye, and it would hang on a wall a grandchild grows up looking at.

Reduce the crease and leave it visible. Two hours, entirely honest, and the family will see the crease every time they pass it.

The studio's owner does this arithmetic often enough to have a rule about it: the decision belongs to the family, and it cannot be made by them unless somebody explains the difference between the three, which takes about four minutes and is the part most studios skip.

Why the order of work was not negotiable

There was a second decision, smaller and entirely technical, about what to do first.

The instinct on a damaged print is to start with the interesting part, which here was the eye. Doing that costs you the work twice. Restoration runs from large damage to small: archive the untouched scan, recover global tone, correct the global colour, then the tears and missing corners, then the creases and stains, then the dust, then local tone, then grain matching, then output.

The reason is that every large repair changes what the small ones have to match. Repair a crease against a faded, pink, low-contrast original and then recover the contrast, and the repair is now visibly a different tone from everything around it. Recover the contrast first and a surprising amount of what looked like catastrophic damage turns out to be faint — on this print, two marks that had been quoted as stains disappeared entirely once the black point was where it belonged.

Working small to large means redoing the small work after every large repair. Working large to small means doing it once.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

He quoted the six-hour option separately and explained why it cost more, showing them the second print and pointing at the eye. The family agreed to the higher price, which he had not expected.

The work ran large damage to small, which is the order that stops you redoing things. Global tone first, to recover contrast from a faded original — a good deal of what had looked like catastrophic damage turned out to be faint once the contrast was back. Then the cast. Then the tear and the missing corner. Then the crease, worked on a frequency-separated pair so the tonal line was evened on the low layer and the skin texture replaced on the high layer, with the lasso stopped short of the jaw where the blur mixes face and background. The eye came last, from the second print, warped and tone-matched, then grain matched across the whole repair.

The colour result was flat, as it had to be. He also made a monochrome version, and that is the one the family framed, which is common enough that he now offers it before it is asked for.

Four files were delivered: the untouched scan at 600 dpi, the colour restoration, the monochrome, and a plain-text note. The note says that the lower left was reconstructed from the surrounding wall, that the corner was cloned from the opposite corner, that the colour cast was corrected but the cyan dye layer had substantially faded so the colours are approximate, and that the grandmother's left eye was rebuilt from the second photograph supplied by the family.

Why is the print pink, and exactly what can and cannot be recovered from that?

The print holds cyan, magenta and yellow dye layers, and in the common chromogenic processes the cyan is the least stable and fades first. What remains — magenta and yellow — reads as red-orange. The correction is therefore to add cyan back, which means pulling the red channel down in the midtones and highlights where the cast is worst. What cannot be recovered is the information the cyan layer held: once that dye has gone there is no weak signal to amplify, so neutralising the cast returns a greyish, low-saturation image. Pushing harder produces channel noise and banding rather than colour, which is why a good monochrome conversion is often the honest and better-looking answer.

Why was the crease treated on a frequency-separated pair rather than healed directly, and where did the lasso have to stop?

A crease is a tonal line over surviving image: the emulsion is cracked but the picture either side is intact. Separating the photograph into a blurred tone layer and a neutral-grey detail layer lets the tonal line be evened on the low layer, where there is no texture to destroy, while the skin texture is replaced on the high layer, where there is no tone to get wrong and therefore no pale disc left behind. The lasso has to stop short of the jawline, because the blur that built the low layer has mixed face and background across that boundary; working on the low layer there produces a visible pale halo tracing the edge of the face.

Rebuilding the eye from the second print and generating one both put pixels there that were not in the original file. What makes them different acts?

Provenance. The rebuilt eye is derived from surviving photographic evidence of that person on that day — transformed, warped and tone-matched, but traceable to a real record. A generated eye is a sample from a distribution of what eyes look like; it is plausible and it is about nobody. The practical difference shows in what can be written down: the delivery note can say which frame the eye came from and what was done to it, and a grandchild can check. For a family photograph on a wall many people would accept the generated version, and the rule is that they should be told that is what it is, which the note makes automatic rather than a matter of remembering.

MODULE 5 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 You heal a cable where a pale wall meets a dark worktop and get a grey smear. Which tool is right and why?

A. Content-aware fill, because it searches the whole frame for a match
B. Generative fill with an empty prompt, so it continues the surroundings
C. The healing brush at a smaller size and lower hardness
D. The clone stamp, sampling along the same side of the edge

2 Why does healing a spot on the high-frequency layer not leave the pale disc that healing the flat photograph sometimes does?

A. The high layer holds no tone for the tool to get wrong
B. The high layer is 16-bit, so the blend is more accurate
C. Linear Light mode hides any tonal error automatically
D. The blur radius has already removed the blemish from that layer

3 A sample jar has a dent in its cardboard box and a scratch on the lid that the production units will also have. What does the governing test for product work permit?

A. Remove both, because the photograph is advertising
B. Remove the dent, keep the scratch
C. Keep both, because the sample is what was photographed
D. Remove the scratch and keep the dent, since packaging is not the product

4 What is the reliable way to check whether a photograph has been reshaped with Liquify?

A. Compare the file size against an unedited frame
B. Check whether the EXIF records the filter
C. Look at straight lines in the background
D. Zoom to 400 per cent and look for repeated texture

5 Which of these is an absence of information rather than a fault in it, so that no tool reverses it?

A. A green cast from fluorescent tubes
B. Barrel distortion from a phone's ultrawide lens
C. A crease across a printed photograph
D. A sky where all three channels read 255

6 A client sends a photograph to be enlarged to A3. Which single check tells you fastest whether the job is possible?

A. Run an AI upscaler and compare it against a bicubic enlargement
B. Open at 100 per cent and look at the most important detail
C. Check the ppi field in the file's metadata
D. Count the layers to see whether it has been edited before

MODULE 6

Compositing: making two pictures one

Putting an element from one photograph into another is easy; making the result believable is a craft with a short list of rules. This block works through the six mismatches that give a composite away, the geometry that tells you where a camera stood, how to relight a cutout and build its shadow and reflection, sky replacement, blending several exposures of one scene, stitching and stacking frames, and the grain and focus matching that removes the last tell.

> BY THE END OF THIS MODULE YOU CAN
>
> Combine elements from different photographs into one image that survives inspection — matching perspective, scale, light direction and quality, edge softness, depth of field and grain — and merge multiple exposures, viewpoints or focus planes of one scene without halos, ghosting or parallax errors

50 · 8 min

Six mismatches, and the eye finds them all

THE ONE THING TO KEEP

Composites fail on six measurable mismatches — perspective, scale, light direction, light quality, edge softness and grain — and the fastest ways to find them are flipping the image, shrinking it and blurring it, because each reveals a different class of error.

Wrong before it is noticed

Show somebody a bad composite and they will say it looks fake before they can say why. That is not vagueness; it is how visual processing works. Consistency of light, perspective and focus is computed early and automatically, long before anything reaches conscious attention, and an inconsistency registers as wrongness with no accompanying explanation.

This is useful, because it means the fix is never "make it look better". It is a checklist of six specific things, each of which can be measured and corrected.

The six

1. Perspective. Was the camera at the same height and the same distance relative to the subject in both photographs? A product shot from slightly above, dropped onto a table shot from eye level, sits wrong no matter how good the cut-out is. This is geometry and it is the next lesson.

2. Scale. Is the element the right size relative to everything else on its ground plane? Scale errors are surprisingly hard to see directly and obvious via shadows and contact points.

3. Light direction. Which way do the shadows fall on the element, and which way do they fall in the plate? A subject lit from the left placed in a scene lit from the right is the commonest failure and the easiest to check — look at the noses, the shadows under objects, and the shaded side of anything round.

4. Light quality and colour. Hard or soft, warm or cool. A subject shot with a small flash has hard-edged shadows with sharp boundaries; an overcast street has shadows so soft they are barely shadows. Mixing them fails even when the direction agrees.

5. Edge quality. Does the element's outline have the same softness as everything else at that distance? A razor-sharp cut-out in a photograph shot at f/2 announces itself immediately, and so does a soft element in a deep-focus scene.

6. Grain, noise and texture. A clean studio element in a grainy night scene reads as pasted, because the surface texture of the two images differs. So does a JPEG-compressed element in a clean raw plate.

There is a seventh that is not about the element at all: **occlusion and contact**. Does the object touch anything? Does anything pass in front of it? An element floating with nothing overlapping it looks pasted even when the six are right, because real objects in a scene are partly hidden by other objects.

The order to work in

Do them in the order that avoids redoing work.

The order that avoids doing the work twice

Geometry first	Colour and exposure, roughly	Light direction and quality	Shadow and contact	Edge softness, then grain
Perspective, scale, position. Everything else depends on where the thing is, and the horizon crossing a standing figure fixes the scale exactly. →	Enough that you are judging the rest against something plausible rather than against an obviously wrong element. →	Relight with gradients clipped to the element, and cool the shade as well as darkening it. A hard-lit element cannot be relit from the opposite side; those shadows are baked in. →	The tight dark contact shadow first. Without it an element has no position at all, and a cast shadow alone still floats. →	Match the depth of field at that distance, then add grain to the whole flattened composite. Grain on the element alone leaves two textures in one picture.

Almost none of the time goes on the cut-out, which is the part beginners expect to be hard. A simple element into a simple plate is an hour; a person into a complex scene with matched light is half a day, and nearly all of it is steps three to five.

1. **Geometry first** — perspective, scale, position. Everything else depends on where the thing is.
2. **Colour and exposure**, roughly, so you are judging the rest against something plausible.
3. **Light direction and quality**, by relighting the element.
4. **Shadow and contact**, which fixes both the floating problem and most remaining scale ambiguity.
5. **Edge quality**, matched to the depth of field.
6. **Grain and noise**, last, applied over the whole composite rather than to the element.

That last point matters: adding grain to the element alone leaves two grain patterns. Adding it to a flattened copy of the whole composite unifies them.

Choosing the plate first

The single largest determinant of whether a composite will work is whether the two photographs were compatible to begin with.

When you can choose, choose:

- **Overcast or open shade** for both, because soft light has no hard shadows to match.
- **Similar focal length**, because that sets the perspective relationship.
- **Similar camera height.**
- **Neutral light colour** in both, which you can then grade together.

When you cannot choose — a client supplies two photographs — the honest assessment comes first. A subject shot with direct sun from the right cannot go into a scene with direct sun from the left. Reversing the element horizontally sometimes solves it, and then any text or asymmetric detail on the subject is mirrored. Say what is achievable before the work starts.

The checking method

When the composite is done, three checks:

Flip it horizontally. Your eye has habituated to the image. Mirrored, it is unfamiliar again and the errors reappear. Flip back afterwards.

Look at a small version. Shrink it to 300 pixels. Errors of tone, contrast and light direction survive; errors of edge detail disappear. If it reads as wrong at 300 pixels, the problem is structural rather than in the mask.

Blur it heavily. A 20-pixel Gaussian on a flattened copy leaves only the broad tonal and colour structure. If the element still stands out as a differently-toned lump, the exposure or colour match is wrong regardless of how good the edges are.

All three take seconds, and all three find different classes of error.

The honest note on effort

A convincing composite is slow. A simple element into a simple plate is an hour; a person into a complex scene with matched light is half a day. Nearly all of that time is items 3 to 6 on the list, not the cut-out, which is the part beginners expect to be hard.

If the schedule does not allow it, the better decision is usually a composite that is *visibly* a composite — a cut-out on a flat or graphic background, which is honest design rather than a failed illusion. The failure mode to avoid is the half-finished attempt at realism, which reads as incompetence where a graphic treatment would have read as a choice.

BEFORE YOU MOVE ON

An editor finishes a composite and it still looks wrong. They flatten a copy, apply a 20-pixel Gaussian blur, and the pasted element is still visible as a distinct lump. What does this tell them?

A. The mask has a hard edge, which survives heavy blurring as a visible boundary.
B. The problem is in the element's overall exposure or colour rather than in its edges, because the blur has removed all edge information and the mismatch remains.
C. The grain of the element does not match the plate, which the blur has averaged into a visible density difference.
D. The element is at the wrong scale, and blurring makes scale errors more obvious by removing distracting detail.

51 · 9 min

The horizon tells you where the camera stood

THE ONE THING TO KEEP

The horizon sits at camera height and crosses every figure standing on the same ground plane at the same point on their body, which fixes an element's scale exactly — and an object's depth is read from where it meets the ground rather than from how large it is.

One line that solves most placement problems

In any photograph taken with the camera level, the **horizon line sits at the camera's height**. Not the middle of the frame — the camera's height above the ground.

That gives you a free measurement. Find something in the plate whose real height you can guess: a person, a door, a chair, a car. Where the horizon crosses them tells you how high the camera was. If the horizon crosses a standing adult at chest level, the camera was at chest height, roughly 1.4 metres. If it crosses at their knees, the camera was low.

And it works the other way. **Every person standing on the same ground plane has the horizon crossing them at the same point on their body.** Two people at different distances, one large and one small in the frame, both have the horizon at their chest. This is the single most useful rule in compositing people into scenes, and it is exact, not approximate.

So the placement procedure becomes:

1. Find the horizon in the plate. Even if it is hidden behind buildings, you can locate it: it is where receding parallel lines converge, and it is the eye level of anyone standing on the ground.
2. Find the same feature on the element. Where does the horizon cross the person in their original photograph?
3. Scale and position the element so the plate's horizon crosses it at the same point.

That gets the scale right without any guesswork about metres.

When the cameras were at different heights

If the element was shot from above and the plate from eye level, the horizon rule tells you they do not match, and no scaling fixes it — you would be looking down at the person and along at the street.

The fixes, in order of quality:

- **Reshoot the element** at the right height.
- **Use it where the mismatch does not show** — put the person further back, where the perspective difference is smaller, or crop to a region where the angle is ambiguous.
- **Perspective-warp the element.** This works for small differences on simple shapes and produces distortion on anything complex, because a warp is a 2D operation being asked to simulate a 3D viewpoint change.

- **Accept it and make the composite graphic** rather than photographic.

Focal length and the shape of things

Focal length does not change perspective — distance does — but the two travel together, because a wide lens is used close and a long lens far away.

What that means for compositing:

- **A wide lens used close** exaggerates the difference in size between near and far parts of the subject. A face photographed at 24 mm from half a metre has a large nose and small ears.
- **A long lens from far away** compresses those differences. Backgrounds look larger and closer relative to the subject.

So an element shot at 85 mm dropped into a plate shot at 24 mm sits oddly, because the internal proportions of the element do not match the perspective of the scene around it. Check the EXIF of both files. A mismatch of 35 mm against 50 mm is usually survivable; 24 mm against 135 mm is not.

Grounding: the contact point decides everything

An object's position in depth is read from **where it meets the ground**, not from its size. Put a cut-out chair with its feet high in the frame and it reads as far away, regardless of how large it is; the eye then judges its size relative to that distance and concludes it is enormous.

This is why the base of an object has to sit exactly on the correct ground line, and why a floating element is not merely missing a shadow — it has no position at all.

For an object that should be partly behind something, **occlusion beats everything**: masking the element so a table edge or a plant passes in front of it does more for believability than any amount of light matching, because occlusion is unambiguous depth information.

Converging lines and the ground plane

When you need to place something on a receding surface — a poster on a wall at an angle, a rug on a floor, a screen on a laptop — the element has to follow the plane's perspective.

The method:

1. Identify two pairs of lines in the plate that are parallel in reality — the top and bottom of a wall, the two sides of a floor.
2. Extend each pair until they meet. That meeting point is the vanishing point for that direction.
3. Transform the element so its own edges point at the same vanishing points.

Photoshop's *Vanishing Point* filter lets you define the plane and then paste into it, doing the transform automatically. GIMP's Perspective transform with the grid overlay does it manually, and the free tool **Hugin** handles extreme cases. In any tool, turning on a grid or dragging guides along the plate's real lines makes this a two-minute job instead of a fiddly one.

Checking

Three tests, all fast:

Extend the lines. Draw a temporary line along the base of the element and along a real edge in the plate. Do they converge to the same point?

Compare heads or tops. For multiple people or objects on the same plane, the line through their tops and the line through their bases should converge at a single point on the horizon.

Look at the element's own ground contact at 100 per cent. A foot half a pixel off the floor is a visible float; a foot overlapping the floor by two pixels reads as sunk into it.

The limit

A photograph is a projection of three dimensions onto two, and the projection is not invertible. You cannot recover the camera's exact position, and you cannot change a photograph's viewpoint after the fact — a warp fakes it convincingly only for small angles on flat or simple surfaces.

Where the two photographs' viewpoints genuinely disagree, the honest answer is that they cannot be combined convincingly, and the work should be redirected: reshoot one, choose a different plate, or design the image so the mismatch is not a claim about a real scene.

BEFORE YOU MOVE ON

A designer composites two people into a street scene. Both are correctly cut out and lit, but one of them looks like a giant. Checking the plate, the horizon crosses the correctly-sized figure at chest height. Where should it cross the other one?

A. At the same proportion of their height as the first figure, since scale in a photograph is proportional rather than absolute.

B. Higher up their body if they are further away, because distant objects sit closer to the horizon line.

C. At chest height too, because everyone standing on the same ground plane is crossed by the horizon at the same height regardless of how far away they are.

D. Anywhere, provided their feet sit on a line that converges correctly with the pavement's vanishing point.

52 · 9 min

Relighting a cut-out so it belongs

THE ONE THING TO KEEP

Read direction, elevation, quality and the key-to-fill colour difference from the plate first, build the tight dark contact shadow before the directional cast shadow, and accept that shadows already baked into a hard-lit element cannot be relit from the opposite side.

Read the plate before you touch the element

Four readings, in two minutes, from the background photograph alone.

Direction. Find any object with a shadow and note which way it points. Find anything round or cylindrical — a pole, a head, a bottle — and note which side is lit. Two independent readings agreeing tells you where the key light is.

Elevation. A short shadow means a high light. A long shadow means a low one. Overhead noon sun gives a shadow roughly a third the object's height; late afternoon gives one two or three times it.

Quality. Look at the edge of a shadow. Sharp edge with a clear boundary: a small or distant hard source, usually direct sun or a bare flash. Soft edge that fades over centimetres: a large source, an overcast sky, a window, a bounce. Somewhere between: partial cloud or a moderately sized source.

Colour. Sample the lit side of something neutral and the shadow side of the same thing. Compare. Under sunlight the lit side is warm and the shadow is blue, because the shadow is lit by sky. Under an overcast sky the difference is small. Under artificial light it can be anything. That difference between key and fill is one of the strongest realism cues there is, and almost nobody matches it.

Relighting the element

Assume the element's light is close enough in direction to be salvageable. Three layers, clipped to the element:

1. A gradient for the key. A Curves layer pulled up, with a gradient on its mask running from the key's direction. This lifts the side facing the light. Keep it subtle; you are adjusting rather than relighting from scratch.

2. A gradient for the shade. A Curves layer pulled down with the opposite gradient, and — this is the part that sells it — the curve's blue channel lifted slightly so the shaded side goes cooler as well as darker. Under daylight that is physically what happens.

3. Dodge and burn, on a grey Soft Light layer clipped to the element, to shape the form: deepen the shadow side of a face or a bag, lift the top surfaces facing the sky.

Where a light source in the scene should be spilling onto the element — a neon sign, a window, a fire — add a colour layer clipped to the element, set to Soft Light or Colour, masked to the side facing the source. A subject standing beside a red wall has red light on that side, and its absence is one of those things nobody notices but everybody feels.

Rim light: the cheapest realism there is

When the key light is behind or to the side, the edge of the subject catches it, giving a bright rim along the outline. Adding one to a composited element does more for integration than almost anything else, because it visually ties the element's edge to the scene's light rather than to the mask.

How: duplicate the element's mask into a selection, contract it by two or three pixels, invert within the original selection to leave a thin band along the edge, then use that as a mask on a Curves layer pulled up. Mask off the band on the side away from the light. Two minutes.

The caution: a rim light on the wrong side is worse than none, and a rim light all the way round reads as a cheap cut-out glow.

Contact shadow versus cast shadow

These are different phenomena, and treating them as one is why so many composites float.

Two shadows that are different phenomena

	What makes it	What is missing without it
Contact shadow	**e object blocks ambient light fro** dark, tight and fairly sharp, and presen	**The object floats** build this one first, before anything directional
Cast shadow	**The key light is blocked by** directional, softening and lighten	**Reads as an overcast day** coloured by the sky and the bounce, never black

Treating them as one is why so many composites float. Build the contact shadow first: it exists regardless of any key light, it is there on an overcast day when nothing casts a shadow at all, and it is the one that grounds the object.

The contact shadow — sometimes called occlusion — is the dark area immediately where the object meets the surface, caused by the object blocking ambient light from every direction. It is **dark, tight and relatively sharp**, it exists regardless of any key light, and it exists even on an overcast day when there is no cast shadow at all.

The cast shadow is the shape thrown by the key light. It is directional, it can be long, and its edge softens with distance from the object.

An object with a cast shadow and no contact shadow still floats. An object with a contact shadow and no cast shadow looks like an overcast day. Build the contact shadow first; it is the one that grounds the object.

The rules a shadow obeys

- **Darkest and sharpest where the object touches**, lighter and softer as it recedes. This is the single most-broken rule in amateur compositing, where a uniformly blurred grey shape is pasted underneath.
- **It takes the shape of the surface it falls on.** A shadow crossing a kerb bends at the kerb. A shadow on grass is broken up by the blades.

- **It is not black.** A shadow is lit by ambient light and by bounce from nearby surfaces. In daylight it is blue; beside a red wall it is reddish. Use a Multiply layer with a colour rather than a black one.
- **It is not uniformly opaque.** 20 to 40 per cent on a Multiply layer is usually right; 80 per cent reads as a hole.
- **Multiple lights make multiple shadows**, each fainter than a single light's would be.

When the directions cannot be reconciled

If the element is lit from the left and the plate from the right:

- **Flip the element horizontally**, if nothing on it is asymmetric or has text. Instant fix.
- **Change the plate** to one whose light agrees.
- **Convert the scene to soft light** by choosing an overcast plate, where direction barely matters.
- **Reshoot.**

What does not work is relighting a hard-lit element to look hard-lit from the other side. The shadows *within* the element — under a nose, inside a fold, beneath a brim — are baked into the pixels, and painting the opposite side bright while those remain gives an object lit from both sides with shadows in the wrong places. Every viewer sees it and none of them can say why.

This is the case where recognising the limit early saves the afternoon, and it is the reason experienced compositors spend so long choosing which two photographs to combine.

BEFORE YOU MOVE ON

A composited handbag has a long, evenly blurred grey shadow stretching away from it, but it still looks as though it is hovering above the table. What is missing?

A. The shadow's opacity is too low, and raising it towards 80 per cent would anchor the object.
B. The shadow should be black rather than grey, since a cast shadow on a lit surface receives no direct light.
C. The contact shadow — the dark, tight occlusion exactly where the bag meets the table — has not been built, and it is that rather than the cast shadow which grounds an object.
D. The cast shadow's direction does not match the plate's key light, and correcting the angle would resolve the floating.

53 · 9 min

Constructing what the light would have done

THE ONE THING TO KEEP

A convincing shadow is dark and tight at the contact and progressively softer and lighter further away, coloured rather than black, and shaped by the surface it falls on — and the easiest way to get one right is to extract the real shadow from the element's own original frame.

Three shadows, built three ways

The contact shadow. On a new layer set to Multiply, paint with a soft black brush at 10 to 20 per cent opacity, tight against the base of the object, following its outline. Two or three passes, darkest right at the join, fading within a few pixels. Alternatively: duplicate the object's mask, contract it, blur it by two or three pixels, and fill the result with a dark colour on a Multiply layer. Either way it is small, dark and close.

The cast shadow from a flat object. Duplicate the object layer, fill it entirely with a dark colour (lock transparency and fill, or use a Colour Overlay), set the layer to Multiply at 25 to 40 per cent, then:

1. *Edit* > *Transform* > *Flip Vertical*, then move it so its top edge meets the object's base.
2. *Distort* or *Skew* it away from the light at the correct angle and length.
3. **Apply a graduated blur**: sharp near the contact point, increasingly soft further away. Do this by applying a Gaussian blur through a gradient mask, or by using *Filter* > *Blur Gallery* > *Field Blur* with two pins at different radii. This step is what separates a real-looking shadow from a pasted grey shape.
4. Mask it where it should be interrupted by objects or fall over edges.

The cast shadow on an uneven surface. Build it as above, then use a **displacement map**: save a blurred greyscale copy of the surface as a PSD, and apply *Filter* > *Distort* > *Displace* to the shadow layer using that file. The shadow now follows the surface's bumps. GIMP has *Filters* > *Map* > *Displace* with the same behaviour. For grass, gravel or cobbles this is what makes it work.

A shadow's density is not one number

The common error is a shadow of uniform opacity. Real shadows have three zones:

- **Umbra** — full shadow, where the light source is entirely blocked. Darkest.
- **Penumbra** — partial shadow, where part of the source is visible. This is the soft edge, and its width depends on the source's size and its distance.
- **Ambient fill** — light reaching the shadow from the sky, from bounce, from other sources. This is why shadows are never black and why they take the colour of whatever is lighting them.

So the realistic construction is a dark, relatively small umbra near the contact, a wide soft penumbra spreading out, and the whole thing coloured rather than neutral. A Multiply layer filled with a dark blue rather than black is closer to right in daylight, and takes no longer.

Reflections on a glossy surface

The standard product treatment, and it takes five minutes.

1. Duplicate the object layer. *Flip Vertical*.
2. Move it so the base of the reflection meets the base of the object **exactly**. A one-pixel gap reads as a floating object; an overlap reads as sunk.
3. Add a layer mask with a **black-to-white gradient**, fading the reflection out downwards. How fast it fades is how matte the surface is.
4. Reduce the layer opacity to 20 to 40 per cent.
5. Add a small **vertical motion blur** or a progressive Gaussian blur increasing downwards, because a real reflection scatters more the further it travels across a surface that is not a perfect mirror.
6. If the surface is textured, apply a displacement map so the reflection follows it.

The details that make it convincing: a reflection is **compressed vertically** when the camera is above the surface rather than level with it, so squash the flipped copy. And it is **darker and less saturated** than the object, because the surface absorbs some light — a Curves layer pulled down on the reflection, plus a small desaturation.

For a reflection in water, add a horizontal ripple: *Filter* > *Distort* > *Ripple* or GIMP's *Filters* > *Distorts* > *Ripple*, at a small amplitude, with the wavelength increasing towards the viewer.

Shadows and reflections from a real frame

The best shadow is a real one. If you shot the element yourself against a plain surface, the shadow is in the original frame.

To extract it: take the original layer, desaturate a copy, invert it, and use it as the mask on a black fill layer set to Multiply. The result carries the real shadow's shape, density gradient and softness, which no construction will match. Then transform it to match the new scene's light direction.

This is the strongest argument for shooting cut-out elements yourself on a white surface rather than sourcing them: you get the shadow for free.

Checking the geometry

A shadow's direction obeys perspective. If two objects sit on the same ground plane lit by the same sun, their shadows are parallel in reality, which means in the photograph they converge on a point — and that point lies on the horizon if the light is level, below it if the sun is behind the camera, above if it is in front.

The practical check, without any of that theory: draw a temporary line along an existing shadow in the plate and along the shadow you have built. Do they agree in direction? Then look at their lengths relative to the objects casting them. A metre-tall object with a two-metre shadow next to a two-metre object with a one-metre shadow is wrong, and it is the sort of wrong that people spot immediately.

The free path and the honest limit

Everything here is available in GIMP, Krita and Photopea: transforms, Multiply layers, gradient masks, displacement maps, ripple and motion blur. Nothing in this lesson needs a subscription.

The limit is that a constructed shadow is a 2D approximation of a 3D interaction. It holds up for a single object on a simple surface with one obvious light. It breaks down for complex objects — a chair with legs and a back, a person with separated limbs — where the real shadow would show internal structure that a flipped silhouette cannot produce, and for scenes with several light sources, where the real result is overlapping shadows of differing densities.

At that point, the two honest routes are to photograph the object on a surface in the real light, or to build it in 3D and render the shadow properly. **Blender**, free, will do that, and for a repeated product task it is often less work than faking it once.

BEFORE YOU MOVE ON

An editor builds a cast shadow by duplicating the object, filling it black, flipping and skewing it, setting Multiply at 35 per cent, and applying a uniform 12-pixel Gaussian blur. It still looks pasted on. Which change would help most?

A. Raise the opacity to 70 per cent so the shadow reads as a real absence of light.
B. Apply the blur through a gradient so the shadow is sharp at the contact point and progressively softer further from the object.
C. Replace the black fill with a dark blue, since daylight shadows are lit by the sky.
D. Add a displacement map so the shadow follows the texture of the surface beneath it.

54 · 8 min

A new sky changes the light in the whole photograph

THE ONE THING TO KEEP

A sky is the scene's light source, so replacing it means matching direction, hardness, exposure and colour spill in the foreground as well — and a plate where the old sky reflects in windows, water or paintwork cannot take a new one without replacing every reflection too.

The commonest composite, and the commonest failure

Estate agency, landscape, architecture, car photography, wedding. A flat white overcast sky is replaced with a dramatic one, and nine times out of ten the result is obviously wrong.

The reason is that a sky is not a backdrop. It is the **light source** for everything else in the frame. Change it and every consistency relationship in the photograph breaks at once.

The four things that must agree

1. Light direction. A dramatic sky with the sun low on the left, over a building whose shadows fall to the left, is a contradiction. Check the shadows in the plate first and choose a sky whose bright area matches.

2. Light quality. The original overcast day produced soft shadows and low contrast on the ground. Drop in a clear blue sky with a hard sun and the ground still has soft shadows. The scene now looks like a bright day in which nothing casts a shadow, which the eye reads instantly even if it cannot articulate it.

3. Exposure relationship. A bright dramatic sky means a lot of light in the scene. If the buildings stay at their original overcast exposure they look dark and dull against it. Conversely, a dark moody sky over a brightly lit foreground reads as fake because the foreground is lit by something the sky does not explain.

4. Colour spill. Everything in the scene is lit by the sky. A golden sunset sky means warm light on every upward-facing and sun-facing surface. A deep blue sky means cool light. Without that, the sky and the ground look like two photographs, which they are.

The method

1. **Choose the sky for the plate**, not the other way round. Look at the shadows, note the direction and hardness, and pick accordingly. This is the decision that determines whether the rest is possible.
2. **Mask the existing sky.** Colour Range on the blue or the white usually gets 90 per cent in one click; the channel technique from earlier handles trees against sky better than anything else. Check the edges around foliage and aerials at 100 per cent.
3. **Place the new sky underneath** and scale it. Watch the perspective — a sky photographed with a wide lens has clouds that recede to the horizon, and stretching it distorts that recession visibly.
4. **Match the horizon.** The new sky's horizon should sit at the plate's horizon, not at the top of the buildings.

5. **Add atmospheric haze at the horizon.** Real skies are paler and less saturated near the horizon because of the depth of air. A subtle white or pale-blue gradient across the bottom of the sky layer is almost always needed and is the single quickest improvement.
6. **Grade the foreground to match.** A Photo Filter or Colour Balance layer over everything below the sky, warming or cooling towards the sky's colour, at low strength.
7. **Add the spill.** A colour layer set to Soft Light, masked to the upward-facing and sky-facing surfaces — roofs, the tops of cars, water, shoulders.
8. **Reconsider the foreground exposure** against the new sky's brightness.

Automatic sky replacement

Photoshop's *Sky Replacement*, Luminar's version and several phone apps do steps 2 to 6 automatically, including an attempt at the colour grade and the spill. They are genuinely good starting points and they mask foliage better than most people do by hand.

What they do not do: choose a sky whose light direction matches your shadows, or fix the light quality mismatch. The automatic grade adjusts colour, not the hardness of shadows that are already in the pixels.

So the automatic tool does the tedious part and leaves you the judgement, which is the correct division of labour. Use it, then check the four agreements.

The free equivalent is to do it manually, which is what this lesson describes, and GIMP handles every step. There is no free one-click sky replacement of comparable quality, and that is an honest gap.

Where the honesty question bites

Sky replacement is a good case study because the answer genuinely depends on the use.

- **Advertising and illustration.** Expected, unremarkable, nobody is misled.

- **Estate agency.** Widely done, and in several markets it has become a point of complaint: a listing showing a house under a blue sky that was photographed on a grey day is describing the photograph rather than the property. Some agency standards now require disclosure. Check what applies where you work.
- **Landscape competitions.** Most major competitions prohibit it, and entries have been disqualified over it.
- **News and documentary.** Prohibited outright. Adding or removing content, including sky, ends careers.
- **Weather, science, evidence.** Obviously not.

The test from earlier applies cleanly: would the viewer be surprised? For a magazine cover, no. For the photograph accompanying a story about a storm, yes.

A line in the delivery note costs nothing: "sky replaced, source file supplied".

The technical limit

Some plates cannot take a new sky. Specifically:

- **Where the sky reflects in the scene** — windows, water, wet roads, car paint. Replacing the sky without replacing every reflection of it leaves the old sky visible in the glass. This is a lot of work and it is where most sky replacements fail on inspection.
- **Where the sky is blown to pure white.** The edge pixels around fine branches are pure white and hold no information about coverage, so the mask has nothing to separate. The result is a halo of white around every twig.
- **Where the subject is backlit** against the old sky, so its edges carry a bright rim from a sun that is now somewhere else.

In each case the honest answer is to reshoot, or to leave the sky alone and improve it with tone instead — a graduated Curves layer can do a surprising amount to a flat sky without replacing anything, and it raises none of the questions above.

BEFORE YOU MOVE ON

A photographer replaces the flat white sky of an architectural shot with a clear blue sky and a low sun. The masking and colour grading are carefully done, yet the result reads as fake. What is the most likely remaining mismatch?

A. The new sky needs atmospheric haze at the horizon, without which it reads as a pasted backdrop.
B. The building was lit by an overcast sky, so it has soft shadows and low contrast that a hard low sun cannot explain.
C. The sky layer is at the wrong scale, and clouds stretched to fit the frame distort the recession towards the horizon.
D. The colour grade was applied to the whole foreground rather than only to the sky-facing surfaces, which flattens the result.

55 · 9 min

Two exposures of one scene, and how not to make it look like HDR

THE ONE THING TO KEEP

Exposure fusion picks the best-exposed pixels and needs no tone mapping, while HDR merge plus tone mapping flattens global contrast and restores it locally, which is exactly why halos and the characteristic grey look appear — and a hand-masked blend of two exposures beats both.

The problem: the scene exceeds the sensor

A room with a window. Expose for the room and the window is pure white. Expose for the window and the room is black. The scene's dynamic range — perhaps 14 or 15 stops — exceeds what a single exposure can hold with detail at both ends.

The answer is more than one exposure. Bracket: three frames at −2, 0 and +2 stops, or five at one-stop intervals for an extreme case. Use a tripod, or at minimum brace the camera, and shoot in rapid succession.

The important detail: **change the shutter speed, not the aperture**. Changing aperture changes depth of field between frames, so the images will not align in sharpness. Most cameras' auto-bracket does this correctly; check.

Two ways to combine them, and they give different results

Exposure fusion picks the best-exposed pixels from each frame and blends them, weighted by how well-exposed each one is. It never builds a high-dynamic-range image at all; it goes straight to a normal-range result. This is what **enfuse** does, and it is what Photoshop's *Merge to HDR* produces when you use the more conservative settings.

HDR merge plus tone mapping first builds a genuine high-dynamic-range file holding the full range in floating point, then compresses that range down to something displayable. The compression is the tone mapping, and it is where the characteristic look comes from.

For most photographs, **fusion gives a more natural result with less effort**. Tone mapping gives more control and more opportunities to produce something that announces itself.

Two ways to combine a bracket, and why one of them has a look

Exposure fusion, as enfuse does it

- Picks the best-exposed pixels from each frame, weighted by how well exposed they are
- Never builds a high-dynamic-range file at all, so nothing has to be tone mapped down
- More natural for less effort, which is why phones do this for every shot
- align_image_stack and enfuse are two free command lines, from Hugin

HDR merge plus tone mapping

- Builds the full range in floating point, then compresses it to something displayable
- The compression flattens global contrast; local contrast is raised to hide that
- Halos at every strong edge, because local contrast compares each pixel to a blurred neighbourhood
- The cure is less of it and a smaller radius, or not tone mapping at all

The grey, halo-ringed appearance is not a fashion, it is the mechanism: reducing global contrast to fit fifteen stops into a display makes the picture flat, and restoring local contrast to compensate puts a bright band along every roofline. A hand-masked blend of two exposures beats both and takes ten minutes.

Why tone-mapped HDR looks like that

The grey, flat, halo-ringed, over-detailed look is not a fashion. It is the mechanism.

To fit 15 stops into a displayable range, a tone mapper has to reduce global contrast substantially — which is why the image goes flat and grey. To stop it looking flat, it then increases **local** contrast, using exactly the blurred-neighbourhood comparison from the tone module. And local contrast at a large radius produces halos at every strong edge, which is why tone-mapped photographs have bright bands around every roofline and dark bands inside them.

Knowing that tells you the cure: **less local contrast, and a smaller radius**. Or avoid tone mapping and use fusion.

The manual method, which is better

The result most professionals actually use, and it takes ten minutes:

1. Load two or three exposures as layers in one document, aligned.
2. Put the one exposed for the shadows at the bottom, the one exposed for the highlights on top.
3. Add a **luminosity mask** to the top layer so it applies only where the underlying image is bright.
4. Refine the mask by hand where needed.

That is it. You are choosing, per tonal region, which exposure to show, with a mask that has no edges anywhere — which is why it produces no halos at all. This is the technique from the luminosity mask lesson doing the job it is best at.

For a window in a room specifically: mask the window exactly (a rectangular selection usually) rather than by luminosity, because the boundary is a real hard edge and pretending otherwise gives a glow around the frame.

Alignment and ghosting

Alignment. Handheld brackets shift between frames. Photoshop's *Auto-Align Layers* handles it; **align_image_stack**, part of Hugin and free, does it from the command line and is excellent:

```
align_image_stack -m -a aligned_ img1.jpg img2.jpg img3.jpg
enfuse -o merged.tif aligned_*.tif
```

That is a complete free exposure-fusion pipeline in two lines.

Ghosting. Anything that moved between frames — leaves, people, cars, water — appears in several positions at once, blended into a smear. Deghosting algorithms pick one frame as the reference for moving regions.

The manual fix is more reliable: pick one frame as the base for the moving area and mask the others away from it. A person walking through a bracketed scene is a mask, not an algorithm.

Free tools

- **Hugin** (align_image_stack) and **enfuse** — alignment and fusion, free, excellent, command line.
- **Luminance HDR** — free, cross-platform, several tone mapping operators including Mantiuk and Fattal. Useful for learning what the operators do.
- **darktable** — has its own HDR merge, and separately its *exposure* and *tone equaliser* modules can recover a remarkable amount from a single raw file, which is often enough that bracketing is unnecessary.
- **GIMP** — blend layers with luminosity masks manually, exactly as described above.
- **Phones** already do this. Every modern phone camera bracket-and-fuses automatically for every shot, which is why phone photographs of backlit scenes look better than a dedicated camera's single frame.

When one raw file is enough

A modern raw file holds 12 to 14 stops. For a scene with 11 or 12 stops of range, a single well-exposed raw plus shadow recovery gives a clean result with no alignment, no ghosting and no bracketing.

The test: expose so the highlights are just below clipping, then lift the shadows in the developer and look at the noise. If it is acceptable, you did not need a bracket. If the shadows are noisy and blotchy, you did — the sensor's noise floor is the real limit, and it is why bracketing still exists on cameras that claim 14 stops.

The honest position

The purpose of combining exposures is to produce an image that looks like the scene looked to you, which a single exposure could not. That is a defensible reason and it is what the eye does continuously by adapting as it scans.

Where it stops being defensible is the same place as everywhere else: when the combination asserts something the scene did not contain. A blended interior showing the real view through the window is honest. A blended interior with a view from a different day is a composite, and the estate agency conversation from the previous lesson applies.

BEFORE YOU MOVE ON

A photographer merges a three-frame bracket of a cathedral interior with an HDR tone mapper and gets bright bands along every column edge and a flat grey overall look. What produced both symptoms?

A. Misalignment between the handheld frames, which shows as edge artefacts and reduced contrast after merging.
B. The frames were bracketed by aperture rather than shutter speed, so the depth of field differs between them.
C. Compressing fifteen stops into a displayable range flattens global contrast, and the local contrast added to compensate creates halos wherever the blurred neighbourhood spans a strong edge.
D. The merged file was 8-bit, so the tonal compression exhausted the available levels and banded along the highest-contrast edges.

56 · 9 min

Many frames, one view, and the pivot point that matters

THE ONE THING TO KEEP

Stitching solves for camera orientation from overlapping features, so it needs 30 to 40 per cent overlap and fails on parallax — rotating anywhere but the lens's entrance pupil makes near and far objects disagree between frames in a way no algorithm can reconcile.

Why stitch at all

Three reasons, and only the first is obvious.

Wider than the lens. The traditional reason: a landscape or an interior that no lens covers.

More resolution than the sensor. Shoot a subject in nine overlapping frames with a long lens and stitch them, and you have a file several times the camera's native resolution, with better per-pixel detail than any single wide shot. This is how large prints of buildings and landscapes are made.

Less distortion. A stitched panorama from a 50 mm lens has the perspective character of a 50 mm lens across a very wide field, without the stretched corners of an ultrawide. For architecture this is a genuine quality advantage.

How stitching works

The software finds **control points** — features visible in two overlapping frames — and solves for the camera orientation and lens parameters that place every frame consistently on a sphere. Then it projects that sphere onto a flat image and blends the seams.

That means two things you can act on. Overlap must be sufficient for features to be found: **30 to 40 per cent between adjacent frames**, more in a featureless sky. And the scene must contain findable features — a blank wall or a plain sky gives the algorithm nothing, which is why panoramas often fail at the top of the frame and succeed along the horizon.

Parallax: the one that actually ruins panoramas

Rotate a camera about the tripod's screw and the lens moves through an arc. Objects at different distances shift relative to each other between frames — **parallax** — and no stitcher can reconcile that, because the frames genuinely disagree about the scene's geometry.

The symptom is unmistakable: distant objects line up and near objects are doubled or sliced, usually along a seam. A railing in the foreground with a ghost of itself.

The fix is to rotate about the **no-parallax point** — the entrance pupil of the lens, which is somewhere inside the lens barrel and is not the sensor or the tripod screw. A panoramic head positions the camera so rotation happens about that point. Finding it takes ten minutes: line up two objects at different distances, rotate, and adjust the camera's position back and forth on the rail until they stop shifting relative to each other.

Without a panoramic head:

- **Distant scenes are fine.** With nothing closer than about twenty metres, parallax is negligible and hand-held panoramas work.
- **Rotate about your own spine**, not by swinging your arms, if hand-holding. It is closer to correct than pivoting the camera.
- **Keep the foreground out** of scenes you intend to stitch without a head.

Projections

The sphere has to be flattened, and the choice changes the picture.

- **Rectilinear** — straight lines stay straight. Correct for architecture. Stretches severely beyond about 100 degrees of field, so it is unusable for very wide panoramas.
- **Cylindrical** — verticals stay straight, horizontals curve. The standard for wide landscape panoramas.
- **Equirectangular** — the full sphere, used for 360-degree virtual tours.
- **Mercator** and others sit between cylindrical and equirectangular and are often the best compromise for a 180-degree scene.

If a stitched panorama looks oddly stretched at the edges, the projection is usually the reason, and switching from rectilinear to cylindrical fixes it in one click.

Shooting for a stitch

- **Manual exposure.** Automatic metering changes between frames and the sky ends up in bands. Meter for the brightest part and lock it.
- **Manual white balance.** Same reason.
- **Manual focus**, locked. Refocusing between frames changes the framing slightly through focus breathing.

- **Shoot in portrait orientation** for a horizontal panorama: more frames, but much more vertical coverage and less need to crop.
- **30 to 40 per cent overlap.**
- **Level the tripod**, or the panorama arcs and you crop away a lot of it.
- **Include a frame of your hand** at the start and end of each panorama. It makes finding the set among 300 files trivial.

Free software, which is the best software here

Hugin is free, open source, and more capable than most paid options. It exposes the control points, lets you add and delete them by hand, offers every projection, and includes tools for exposure correction across frames and for masking out moving objects. Its optimiser can solve for lens distortion from the images themselves, which means it produces better results than software relying on a generic profile.

It is not friendly. The assistant tab does most jobs in three clicks; the rest of the interface is a photogrammetry application and looks like one. It is worth the afternoon.

Underneath it, **enblend** does the seam blending and **enfuse** the exposure fusion, both usable independently from the command line.

Photoshop's *Photomerge* and Lightroom's panorama merge are easier and adequate for straightforward cases. Phones stitch in-camera and do it surprisingly well for casual use.

What fails

Moving subjects. A person who walks through two frames appears twice, or half of them appears once. Hugin lets you mask which frame wins in a given region, which is the reliable fix.

Water and foliage. Waves and leaves move between frames, so the seam through them never matches perfectly. Place seams away from them where you can.

Close foregrounds without a panoramic head, as above.

Featureless regions. A plain sky or a blank wall gives no control points. Hugin lets you add them manually, or you can constrain the solution using the known geometry.

Scenes where the light changes during the sequence. A cloud crossing the sun between frame two and frame seven produces a band you cannot blend away. Shoot fast, and reshoot if the light moves.

The limit

A panorama is a single viewpoint rendered wide. It is not a wider scene — you cannot see round a corner, and nothing hidden behind a foreground object is revealed. And every projection distorts something, because a sphere cannot be flattened without distortion. Cartographers have known this for four hundred years, and the choice of projection is the same choice they make: which distortion you are prepared to accept.

BEFORE YOU MOVE ON

A hand-held panorama of a garden stitches cleanly along the distant hedge but the foreground fence posts are doubled and sliced along one seam. What went wrong?

A. The overlap between adjacent frames was insufficient, so the stitcher found too few control points on the fence.
B. The camera rotated about a point other than the lens's entrance pupil, so near and far objects shifted relative to each other between frames.
C. The projection was set to rectilinear, which stretches the frame edges and duplicates detail near the seams.
D. Automatic exposure changed between frames, and the blending algorithm compensated by overlaying both versions of the fence.

57 · 8 min

When depth of field is not enough

THE ONE THING TO KEEP

Focus stacking picks each pixel from the frame where local contrast is highest, so it needs frames whose sharp zones overlap and pixel-accurate alignment including the scale change from focus breathing — and it produces halos exactly where a sharp foreground edge meets a background sharp in a different frame.

The problem, with numbers

Depth of field shrinks dramatically as you get closer. At a 1:1 macro magnification at f/8, the zone of acceptable sharpness is under a millimetre. An insect's eye is sharp and its wingtip is not, and there is no aperture that fixes it — stopping down to f/22 buys a little more depth and loses sharpness everywhere to diffraction.

The same problem, less extreme, appears in product photography: a bottle photographed from close range has a sharp label and a soft neck. And in landscape, where a foreground rock a metre away and a mountain at infinity cannot both be sharp at a usable aperture.

Focus stacking takes several frames at different focus distances and combines the sharp parts of each.

Shooting a stack

- **Tripod, always.** The frames must align to the pixel.
- **Manual focus, moved in small steps** between frames, from nearest to furthest. The steps must be small enough that the sharp zones overlap; if they do not, you get a band of softness through the middle of the result.
- **Do not change the aperture.** Depth of field must be consistent.

- **Move the focus, not the camera**, unless you have a macro rail — moving the camera changes the composition and the perspective.
- **How many frames:** a product shot might need 5 to 15, a macro subject 30 to 100. Many cameras now have an automatic focus bracketing mode that takes the whole sequence for you.
- **Use a remote or the self-timer** so nothing shifts between frames.

Focus breathing, and why alignment is not optional

As the lens refocuses, its effective focal length changes slightly. The frame therefore **changes magnification** across the stack — a phenomenon called focus breathing — so the last frame is slightly more or less magnified than the first.

That means the frames do not simply overlay; they have to be scaled as well as shifted. Every stacking tool does this, and it is why the outer edges of a stacked image are unusable and get cropped.

```
align_image_stack -m -a aligned_ -v *.tif
enfuse --exposure-weight=0 --saturation-weight=0 --contrast-weight=1 \
       --hard-mask -o stacked.tif aligned_*.tif
```

The `-m` flag lets align_image_stack correct for scale as well as position, which is exactly the focus-breathing correction. The enfuse weights tell it to choose pixels by local contrast — sharpness — rather than by exposure, which is the focus-stacking mode. Both tools are free and part of Hugin.

How the combination works, and where it goes wrong

The algorithm computes local contrast at every pixel in every frame and picks the frame where it is highest, because the sharpest version of a region has the most local contrast. Then it blends the choices to avoid hard transitions.

The failure follows directly from that. **Where a sharp foreground edge sits against a background that is sharp in a different frame, the algorithm has to choose between two regions that are both locally**

high in contrast. What you get is a halo: a ring of blurred or doubled detail tracing the foreground object's outline.

This is most visible on:

- Thin structures — antennae, hairs, stamens, wires — where the blurred version in other frames is wider than the sharp one, so the blur survives around the sharp core.
- Objects against a bright background, where the out-of-focus version of the object was larger and lighter.

The fixes: more frames with smaller focus steps, so the transitions are gentler; hand-retouching the halo afterwards from one of the source frames; and, in Photoshop's stacking, adjusting the blend or masking manually. Photoshop's *Edit > Auto-Blend Layers* with *Stack Images* does a reasonable job and leaves the masks editable, which is its main advantage over a command-line tool.

Movement kills it

Anything that moves between frames appears in the wrong place in the frames where it was sharp. A breeze on a flower over a 40-frame stack is fatal. So:

- Work indoors, or on a still day, or shield the subject.
- Shoot the stack as fast as the equipment allows.
- For living subjects, a stack is usually impossible, which is why macro photographers of insects work early in the morning when the subject is cold and still, and why some subjects are only ever photographed in a single frame.

Free software

- **Hugin's align_image_stack + enfuse**, as above. Free, scriptable, excellent.
- **darktable** can align and blend, though it is not its primary purpose.
- **GIMP** with the G'MIC plugin has a focus-stacking filter.
- **Photoshop** does it with Auto-Align plus Auto-Blend, and is the easiest if you have it.

- Several phones now do a two-frame near-far blend automatically in macro mode.

The honest position

Focus stacking is not misrepresentation in most contexts. It produces an image of the subject as it is, with more of it in focus than one exposure could show — much as your eye does by refocusing as it scans.

Where it needs care is scientific and documentary work, where the convention is to disclose it, and competition entries, where rules vary and some categories prohibit any multi-frame technique. Natural history competitions in particular have differing rules on stacking, so check before entering rather than after.

And the real limit: stacking makes everything sharp, which is sometimes worse. Depth of field is a compositional tool — it separates subject from background and directs attention. A fully sharp macro photograph can look flat and clinical, like a specimen, because the visual hierarchy that focus provided has gone. Stack when the whole subject needs to be legible; do not stack by default.

BEFORE YOU MOVE ON

A macro photographer stacks 30 frames of a flower. The result is sharp throughout, but every stamen has a soft ghost outline around it. What causes this?

A. The focus steps were too large, so the sharp zones did not overlap and a band of softness fell across the stamens.

B. The alignment did not correct for focus breathing, so the frames are at slightly different magnifications and thin structures are doubled.

C. In the frames where a stamen is out of focus its blurred image is wider than its sharp one, so the surrounding blur is still the highest-contrast content in that region and survives the selection.

D. The blending used exposure weighting rather than contrast weighting, so brightness rather than sharpness decided which frame won at each pixel.

58 · 8 min

The last tell: texture and depth

THE ONE THING TO KEEP

The last tells are surface ones — depth of field, grain, lens artefacts, compression and black point — and they are fixed by applying a single unified treatment across the whole flattened composite rather than correcting the pasted element on its own.

Why a perfect cut-out still looks pasted

Everything about the element is correct — geometry, light, shadow, colour — and it still reads as added. What is left is **surface**: the grain, the noise, the softness, the lens artefacts that the plate has and the element does not.

These are the cues nobody articulates and everybody registers. Matching them is the last half-hour of a composite and it is what separates professional work.

Depth of field, matched

The first and largest. In the plate, things at different distances have different sharpness. An element placed at five metres must have the sharpness of five metres, not the sharpness of the studio shot it came from.

Measure it. Find something in the plate at approximately the element's intended distance and look at its edges at 100 per cent. Is it crisp? Slightly soft? Clearly blurred?

Apply it. A Gaussian blur on the element, usually 0.5 to 2 pixels, is enough for most placements. For an element that spans depth — a person standing at an angle, a table receding — apply the blur through a gradient mask so the near part stays sharper.

A better match than Gaussian blur for a background element: **Lens Blur** (*Filter* > *Blur Gallery* > *Lens Blur*), which reproduces the shape of the aperture's bokeh, so bright points become discs rather than smears. GIMP has *Filters > Blur > Lens Blur* with a similar aperture-shape control. The difference shows anywhere there are small bright highlights.

The commonest error is the reverse: a sharp element in a soft scene. A razor cut-out of a product placed on a background shot at f/2 announces itself in a fraction of a second.

Grain and noise

Every photograph has a noise floor. A studio element shot at ISO 100 is far cleaner than a street plate shot at ISO 3200, and placing one in the other gives you two textures in one picture.

Measure the plate's noise. Zoom to 200 per cent on a flat area — a wall, a sky, a road surface. How coarse is the grain? Is it monochrome or coloured?

Match it. The reliable method:

1. Finish the composite.
2. Stamp visible onto a new layer at the top (Ctrl/Cmd+Alt+Shift+E).
3. Above that, add a new layer filled with 50 per cent grey, set to **Overlay** or **Soft Light**.
4. *Filter > Noise > Add Noise*, Gaussian, Monochromatic, at 1 to 4 per cent.
5. Adjust the layer opacity until the grain matches the plate's.

Applying it over the whole composite rather than to the element is the key step. Two grain patterns is the problem; one grain pattern across everything is the solution.

For a coarser, more film-like grain, apply the noise to a layer at half resolution and scale it up, which gives larger grain clumps than the per-pixel filter produces. GIMP's *Filters > Noise > RGB Noise* and G'MIC's film-grain filters both do this well and free.

Lens artefacts as realism cues

The plate was taken through a lens, so it has vignetting, chromatic aberration and a particular sharpness falloff towards the corners. An element pasted in has none of those, or has a different lens's version of them.

- **Vignetting.** Apply a gentle darkening at the corners of the *whole composite*, not the element. This unifies everything under one lens.
- **Chromatic aberration.** A tiny amount — half a pixel of red-cyan offset — added to a composited element that sits away from the frame centre can be what makes it sit in the picture. This is genuinely used in visual effects work and it feels perverse the first time.
- **Slight corner softness**, matching the plate's.

Compression artefacts

If the plate is a heavily compressed JPEG and the element is from a clean raw, the plate has visible 8×8 block structure and the element does not.

The fix is unpleasant and correct: degrade the element to match. Save the finished composite as a JPEG at a quality that produces similar artefacts, or add a subtle blocking by scaling a copy down and back up. A clean element in a blocky plate is as obvious as a sharp element in a soft one.

This is another argument for delivering a composite as one re-encoded file rather than assembling from mixed-quality sources and hoping.

Colour and contrast, one last pass

A grading layer over the whole composite unifies more than any per-element correction.

- A single Curves or Colour Balance layer at the top, applied to everything.
- A subtle overall colour cast — even two or three points — ties disparate sources together, which is part of why films are graded.
- Matching black points matters: if the element's darkest pixel is 4 and the plate's is 22, the element has more contrast and will read as separate. Lift

the element's blacks to meet the plate's.

That black-point match is worth singling out, because it is quick, measurable with the eyedropper, and frequently the difference between a composite that works and one that does not.

The checking routine, once more

1. **Fit to screen**, and toggle the element off and on. Does anything jump?
2. **100 per cent along the edge**, the whole way round.
3. **Flip horizontally.** Errors reappear.
4. **Shrink to 300 pixels.** Structural errors survive.
5. **Leave it overnight** and look again. This is not a joke; visual habituation is real and an hour with the file makes you the worst available judge of it.

The honest limit

All of this makes a composite convincing to a viewer looking at a picture. It does not make it undetectable by analysis — noise statistics, compression history, lighting inconsistency and sensor pattern all leave traces that forensic tools can find, and the field of image forensics exists precisely because of it.

That matters for a specific reason: the convincingness of a composite is not a measure of its honesty. A picture that fools the eye is doing exactly what it was made to do, and whether it should exist is answered by what the picture is for, who will see it and what they will believe about it — which is the same question this course has asked at every stage, and the only one no technique answers.

BEFORE YOU MOVE ON

An editor adds film grain to a composited element so it matches the noisy street plate it sits in. At 100 per cent the element now looks grainy, but the composite still reads as two pictures. Why?

A. The grain was added as monochromatic noise, while the plate's noise is coloured, so the two textures differ in character.
B. The element's grain is a second, independent grain field laid over one region, so the picture contains two textures rather than one continuous one.
C. Grain must be applied beneath the element rather than above it, or it sits on the surface instead of in the image.
D. The grain amount was too low to match a high-ISO plate, and raising it to 8 to 10 per cent would unify them.

CASE STUDY · MODULE 6

The blue sky that was still grey in the windows

An estate agency in Thane with eleven flats to list before the end of the month and a week of monsoon cloud.

The photographs they had

A four-storey building with glass balcony fronts, a tiled forecourt still wet from the morning, and two cars parked along the near side. Shot on a flat white overcast day, because that was the day the keys were available. The agency's house style, like most of its competitors', puts a blue sky behind every exterior.

The junior who had been given the files ran an automatic sky replacement on the first one. The masking round the neem tree at the edge of the frame was better than he would have managed by hand in an hour, and the result was obviously wrong in a way he could not immediately name.

It was wrong in the way a sky replacement is always wrong when nobody checks the four agreements.

Direction. The new sky had a bright area low on the right. The building's soft shadows, such as they were, did not agree with anything, because an overcast day produces no directional shadow at all.

Quality. This is the one that does the damage. The original light was soft, so nothing in the frame casts a hard shadow. Put a clear sky with a visible sun behind it and the scene becomes a bright day in which nothing casts a shadow, which every viewer registers instantly and almost nobody can articulate.

Exposure relationship. Against a bright sky, the building stayed at its overcast exposure and read as dull and slightly grey.

Spill. Nothing in the scene had been lit by the new sky. Roofs, the tops of the cars and the wet forecourt all carried the colour of the old one.

And then the thing that actually killed it: **the old sky was still in the building.** Grey, in every balcony pane, in the puddles on the forecourt, and along the top of both cars.

Three routes

Replace properly. Choose a sky whose light matches the plate rather than the other way round, add horizon haze, grade the foreground, add spill — and then replace the reflection in every pane of glass, every puddle and two car roofs. That is the better part of a day per photograph, and eleven listings were due.

Improve the sky that is there. A graduated Curves layer over the existing flat sky, twenty minutes, no new content, no reflections to chase. The sky stays grey.

Wait for a clear morning. The photographs are correct and the listing goes live a week later, in a market where the first week carries most of the enquiries.

The check that would have taken thirty seconds

The junior had looked at the replaced sky at fit-to-screen for ten minutes and knew something was wrong without being able to name it. Three checks would have named it, and all three are seconds of work.

Flip it horizontally. The eye habituates to an image quickly, and a mirrored version is unfamiliar again, so errors that had stopped registering reappear. Flip back afterwards.

Shrink it to 300 pixels. Errors of tone, contrast and light direction survive at that size and errors of edge detail disappear. If it still reads as wrong small, the problem is structural rather than in the mask — which is what it was here.

Blur it heavily. Twenty pixels of Gaussian on a flattened copy leaves only the broad tonal and colour structure. The building stayed a dull grey lump under a bright sky, which is the exposure relationship failing, and it was obvious in a way it had not been at full size.

None of those checks needs experience. They need somebody to have been told that they exist.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

The agency's principal made the call, and she made it per photograph rather than as a policy.

For the eight flats where the interiors were the subject and the exterior shot showed little sky, they used the graduated Curves version: twenty minutes each, no reflections involved, and nobody has ever asked about them. For the three where the building itself is the selling point, they waited for a clear Saturday and reshot. The listings for those went live six days later than planned.

The one photograph that had already been replaced and published was pulled after a viewer, standing in the flat, pointed at a balcony pane in the listing photograph and asked why the sky in it was grey. That conversation cost more than the week.

The agency added one line to its listing descriptions — the date the photographs were taken, and a note that skies are not replaced — initially as a defensive measure and then because two sellers said it was why they had chosen the firm. What was intended as a limitation turned out to be the only thing in the listing that could be checked.

Name the four things that must agree when a sky is replaced, and say which of them an automatic tool cannot fix.

Light direction, light quality, the exposure relationship between sky and ground, and colour spill onto everything the sky lights. Automatic tools do a good job of the masking and make a reasonable attempt at the colour grade and the spill, both of which are adjustments to values. What they cannot fix is light quality: the softness or hardness of the shadows is already baked into the pixels of the foreground, and no adjustment makes a soft shadow hard. That is why the sky has to be chosen for the plate rather than the plate adjusted to the sky, and choosing is a judgement the tool does not make.

Why did the reflections defeat this composite, and what would fixing them have cost?

A sky is the scene's light source, so its image appears wherever the scene is reflective — glass, water, wet paving, car paint. Replacing the sky above the roofline while the old grey sky is still visible in every balcony pane leaves a contradiction that survives every other correction, and it is exactly what a viewer standing in the flat will compare against reality. Fixing it means masking and replacing each reflective surface separately, with the new sky's colour and brightness, distorted by the geometry of each pane and each puddle — most of a day per photograph, on eleven listings due the same month.

Is sky replacement acceptable in estate agency work? Give the test rather than a verdict.

The test is whether the picture is a record or a construction, which is determined by its use rather than by how it was made. A photograph on a poster advertising a development is a construction and nobody is misled. A listing photograph is offered as a record of a specific property, and a buyer treats it as evidence of what they would be buying, which is why sky replacement has become a point of complaint in several markets and why disclosure conventions are emerging. The working question — would the viewer be surprised to learn what was done — answers it here, and a line in the listing costs nothing.

MODULE 6 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 You are placing a photograph of a standing person into a street plate. What fixes their scale exactly rather than approximately?

A. Matching their height against the doorway behind them
B. Matching the focal length recorded in both files' EXIF
C. The horizon crosses everyone on the ground alike
D. Scaling them until their head sits on the upper third line of the frame

2 A composited box has a long soft cast shadow and still looks as though it is hovering a centimetre above the table. What is missing?

A. A rim light along the edge of the box facing the key light
B. The dark shadow where it meets the surface
C. A reflection of the box in the floor beneath it
D. Matching grain between the box and the plate

3 A building with glass balconies, photographed on a grey day, resists sky replacement. Which problem cannot be solved by grading the foreground?

A. The foreground exposure is too dark against the new sky
B. The colour spill on the roofs still comes from the old sky
C. The new sky's horizon sits too high in the frame
D. The old sky is still reflected in the glass

4 Why does heavily tone-mapped HDR have bright bands along every roofline?

A. Local contrast compares each pixel to a blur of its area
B. The merge aligned the bracketed frames imperfectly
C. The floating-point file was converted to 8-bit too early
D. The bracket was shot by changing aperture rather than shutter

5 A hand-held panorama of a courtyard stitches perfectly in the distance and doubles a foreground railing along one seam. What went wrong?

A. There was not enough overlap between adjacent frames
B. Exposure was on automatic and changed between frames
C. The camera did not rotate about the entrance pupil
D. The projection was rectilinear rather than cylindrical

6 You have added grain to a composited element to match a noisy plate, and it still reads as pasted. What is the fix?

A. Add more grain until the two textures agree
B. Apply a lens blur to the element so the grain softens
C. Denoise the plate down to the element's noise level
D. Add grain to the whole flattened composite instead

MODULE 7

AI inside the image editor

Generative fill, upscaling, denoising and colourising are now buttons in ordinary editors, and every one of them produces content rather than recovering it. This block explains what the models are doing so their failures become predictable, shows how to run the free equivalents on a laptop with no graphics card, covers provenance and watermarking as they actually stand, and ends on the decision about when none of it should be used.

> BY THE END OF THIS MODULE YOU CAN
>
> Use generative fill, outpainting, upscaling, denoising and colourising deliberately — predicting each one's characteristic failures from its mechanism, running free local equivalents where confidentiality or cost requires it — and state for a given image whether generated content is permissible and what has to be disclosed

59 · 10 min

Noise in, picture out: inpainting explained

THE ONE THING TO KEEP

Generative fill initialises a masked region with noise and denoises it towards something consistent with the surrounding pixels and your prompt, which is why filled areas are cleaner than photographs, softer than the original file, and wrong wherever text or regular geometry is involved.

The training idea, in one paragraph

A diffusion model is trained by taking real images and adding noise to them in steps until nothing remains but static. The model learns to predict, at any noise level, what noise was added — which is equivalent to learning how to remove a little of it. Run that prediction repeatedly starting from pure static and you get an image, because each step nudges the static towards something the model considers image-like.

Text conditioning steers the nudging. A text encoder turns your prompt into a vector, and the denoising network is trained so that its predictions depend on that vector. "A ceramic bowl" pushes each step towards arrangements the model associates with those words.

Inpainting: the same thing with a mask

When you select an area and ask for generative fill, four things go in:

1. **The masked region**, which is initialised with noise.
2. **The surrounding pixels**, which are kept fixed and re-imposed at every denoising step, so the generated area is continuously pulled towards agreeing with its edges.
3. **Your prompt**, if you gave one.

4. **A denoising strength**, which decides how much of the original content under the mask survives. At 1.0 the region is generated from scratch; at 0.4 the original is only partially overwritten, which is useful for changing a texture while keeping a shape.

Most systems work in a **latent space** rather than on pixels: an encoder compresses the image by a factor of eight in each direction, the diffusion happens in that smaller space, and a decoder expands the result back to pixels. This is why the whole thing is fast enough to be a button, and it is the source of several of the artefacts below.

What an inpaint is actually given

The masked region, initialised with noise
Nothing of the original survives at full denoising strength. Around 0.4 the shape underneath still constrains the result, which is how a texture is changed and a fold kept.

→

The surrounding pixels, re-imposed at every step
This is what pulls the generated area towards agreeing with its edges, and it is the whole reason an empty prompt continues your wall rather than inventing a generic one.

→

Your prompt, as a vector
Short and specific beats long and elaborate. Long stacked style words dilute the conditioning rather than compounding it.

→

Twenty to fifty denoising steps, at one-eighth scale
Diffusion happens in a latent grid eight times smaller in each direction. Fine detail is decoded rather than generated, which is the resolution ceiling.

→

A decoded patch that is cleaner than your photograph
Likely images are not grainy, so the fill has no noise floor. Match grain over the whole picture afterwards, and expect text and brick courses to be wrong.

The surroundings are a far stronger constraint than anything you type, which is why an empty prompt is right for a removal and why a mask drawn tight to the object leaves a seam. Expand the selection by 10 to 30 pixels and feather 5 to 15, and most of the complaints about the tool disappear.

What follows from the mechanism

Generated regions are smoother and cleaner than photographs. The model produces what it considers a likely image, and likely images are not grainy. A filled region in a high-ISO photograph is conspicuously noise-free. Add matched grain over the whole picture afterwards, as in the compositing block.

The latent grid produces a resolution ceiling. Because diffusion happens at one-eighth scale, fine detail is decoded rather than generated, and the result is often slightly soft compared with the surrounding photograph. On a 45-megapixel file with a small fill, the difference is obvious.

Seams appear when the mask boundary is too tight. The surrounding context is what keeps the generation consistent, and a mask drawn exactly at the edge of the object gives the model almost no overlap to work with. Expand the selection by 10 to 30 pixels and the seam usually disappears.

Text comes out wrong. The model has learned the visual statistics of letter-forms, not a writing system. A generated shopfront has letter-shaped marks that are not letters. This has improved and it has not been solved, and for anything with real text the answer is to composite the real text in.

Regular geometry drifts. Brick courses that do not line up, window frames that change width, tiles that lose their grid. The model has no representation of a straight line continuing; it has a statistical sense of what brickwork looks like locally.

Hands, small faces and reflections are the classic failures, all for the same reason: they are high-complexity structures that occupy few pixels, so errors there cost the training objective very little.

It is not deterministic. The same prompt on the same mask with a different random seed gives a different result. Three attempts cost three seconds and one of them is usually much better than the others, which is why every implementation offers variations.

The controls worth learning

- **Mask feather and expansion.** The single most effective control. Feather 5 to 15 pixels, expand 10 to 30.
- **Denoising strength**, where exposed. High for replacing, low for modifying.
- **Prompt or no prompt.** An empty prompt tells the model to continue the surroundings, which is what you want for removing an object. A described prompt is for adding something.
- **Generate in several small passes** rather than one large one. Each pass has more surrounding context relative to its size, so each is more consistent.
- **Seed**, where exposed, so you can reproduce a result you liked.

Where this sits against the earlier tools

The previous module's content-aware fill copies patches from your own photograph. Diffusion inpainting generates new content from a model trained on other people's photographs. The practical difference:

- **Content-aware** is limited to what is in the frame, so it cannot invent a doorway — and everything it produces really was in your image.
- **Diffusion** can produce a doorway, and that doorway was never there.

Neither is better in general. For removing a bin against a plain wall, content-aware is faster and safer. For removing a car from a complex street, diffusion will produce a more convincing result and will have invented the shopfront behind it.

What it does not do

It does not know what is true. The model has no representation of your client's product, this particular person's face, the actual building behind the van, or what was written on the sign. It produces the most probable-looking content given the surroundings and the prompt, and probability is not evidence.

That statement is the whole of the ethical discussion later in this module, and it is worth holding as a technical fact rather than a moral one: **the output is a sample from a distribution, not a measurement of a scene.** Everything else follows from deciding whether the picture in front of you is allowed to contain samples.

BEFORE YOU MOVE ON

An editor removes a lamp post with generative fill, drawing the selection tightly around it. The fill matches the background but there is a visible seam. What is the most effective fix?

A. Raise the denoising strength to 1.0 so the region is generated entirely rather than partly derived from the lamp post.
B. Write a more specific prompt describing the background, so the model produces content that matches more closely.
C. Expand and feather the selection so the model has 10 to 30 pixels of surrounding context to reconcile the generated region against.
D. Generate at a higher output resolution, since the seam is caused by the latent space's eightfold downscaling.

60 · 9 min

Getting a usable result in three attempts instead of thirty

THE ONE THING TO KEEP

Leave the prompt empty when removing and describe the light and material when adding, generate three variations and composite the good parts onto a masked layer rather than accepting a whole result, and fix grain, geometry and black point afterwards.

Decide what you are asking for

Three different jobs, three different approaches.

Removing something. Leave the prompt **empty**. An empty prompt tells the model to continue what surrounds the mask, which is exactly what you want. Typing "grass" when removing a bin from a lawn often makes things worse, because it biases the generation towards a generic idea of grass rather than towards *this* lawn.

Adding something. Describe it, including the light and the material. "A terracotta plant pot, soft window light from the left, shallow depth of field" produces a more usable result than "plant pot", because those words also steer the rendering rather than only the object.

Changing something. Use a moderate denoising strength if the tool exposes one, so the existing shape constrains the result. Changing a shirt's pattern while keeping its folds is this case.

The practical settings

- **Mask expansion of 10 to 30 pixels and a feather of 5 to 15.** As the previous lesson explained, this is the setting that matters most.
- **Work in passes.** Fill the largest region first, then run smaller fills over the parts that came out wrong. Each subsequent pass sees the previous one as context, so the result converges.
- **Generate three variations, always.** They are cheap and the spread is large.
- **Composite rather than accept.** Put the generated result on its own layer, mask it, and keep the parts that worked. You are not obliged to take the whole thing.

That last point is the one that separates people who get good results from people who conclude the tool does not work. A generation that is 70 per cent right and 30 per cent wrong is a useful layer, not a failure.

A worked example

A photograph of a table with a coffee cup, and the client wants the cup gone and the wooden surface continuous.

1. Select the cup with a rough lasso, expand by 20 pixels.
2. Empty prompt. Generate three.
3. Two produce plausible wood; one has a ghost of a shadow. Take the best.

4. Put it on a layer, mask it to just the cup area plus a small margin, and check the wood grain lines against the real ones either side. They probably do not continue.
5. Fix the grain: clone a strip of real grain from elsewhere on the table, transform it to the right angle, and mask it over the discontinuity.
6. Add grain and match the black point across the whole image, as in the compositing block.

Steps 4 to 6 are what make it invisible, and they are the steps the demonstration videos leave out.

Resolution, and the two-pass technique

Most implementations generate at a fixed internal resolution and scale the result to fit your selection. On a large file that means a soft patch.

The workaround: **crop, fill, and composite back.** Copy a region around the area you want to fill into a new document at a size close to the model's native resolution, do the fill there, then bring the result back and blend it in. More steps, considerably sharper.

Or fill at the original resolution and then sharpen the filled region locally to match, which is cruder and often enough.

Where it beats manual work, and where it does not

Faster than manual: organic textures — foliage, gravel, water, sky, crowds, grass, clouds. Extending a background. Removing a person from a complex scene. Filling a corner of a badly-shot frame.

Slower than manual, or worse: anything with text. Anything with regular geometry — brickwork, tiling, panelling, windows. Small faces. Hands. Product surfaces, where the result must be the actual product. Repeated patterns where a copy-and-transform is exact and a generation is approximate.

Do not use at all: where the content matters as a record. A document, a number plate, a medical image, a news photograph, a forensic exhibit.

Prompting notes that are actually useful

- **Describe the photograph, not the object.** "Shot on 50 mm, overcast light, slight film grain" steers the rendering towards something that will composite.
- **Negative prompts**, where available, are for the recurring failures: "text, watermark, extra limbs, duplicate".
- **Short and specific beats long and elaborate.** Long stacked prompts of style words dilute the conditioning rather than compounding it.
- **The surrounding pixels dominate.** For an inpaint, the context is a far stronger constraint than the prompt, which is why an empty prompt works so well for removals.

The free routes

Adobe's Generative Fill is in Photoshop and requires a subscription and generative credits. The free equivalents run locally:

- **ComfyUI** or **AUTOMATIC1111's web UI** with an inpainting model. Both free, both run Stable Diffusion variants locally. With a graphics card they are fast; on a CPU an inpaint takes one to several minutes, which is slow but workable for a handful of fills.
- **Fooocus** is a much simpler front end with an inpaint mode and sensible defaults, aimed at people who do not want to learn a node graph.
- **Krita** has an AI diffusion plugin that puts local inpainting directly into the editor, on a layer, with masks — which is the most integrated free option there is.
- **GIMP** has several plugin routes of varying maintenance quality.

The hardware reality is covered later in this module. The short version: 8 GB of video memory runs these comfortably, 4 GB runs them slowly, and no graphics card runs them very slowly but does run them.

The limitation to keep in front of you

A generative fill produces something that looks right. Whether it *is* right is a question the tool cannot answer and does not attempt to.

For a lawn, nobody cares. For a shopfront, a label, a face or a road sign, the difference between looking right and being right is the entire content of the picture. The practical habit is to ask, before each fill: if this region were wrong, would anyone be misled? If yes, do it by hand or leave it.

BEFORE YOU MOVE ON

An editor removing a bin from a lawn types the prompt "green grass" and gets a patch that looks like grass but does not match the lawn's mowing stripes or colour. Why does an empty prompt usually work better here?

A. An empty prompt disables the text encoder, which halves the computation and leaves more capacity for detail.

B. With no prompt, the generation is steered only by the surrounding pixels, so it continues this lawn rather than a generic idea of grass.

C. Text prompts are only supported for adding objects, and using one for a removal produces undefined behaviour.

D. The prompt should have been more detailed, describing the stripes and the light, which would have matched the lawn precisely.

61 · 8 min

Making a photograph bigger than it was taken

THE ONE THING TO KEEP

Outpainting is inpainting with the mask outside the frame, so it works best in 20 to 30 per cent stages with generous overlap — and because viewers read extended content as part of the scene, it is the generative operation most in need of disclosure outside advertising and design.

The job this solves

A photograph shot in portrait is needed as a wide banner. A square product shot needs to become 16:9 for a slide. A portrait cropped too tight needs headroom for a magazine's masthead. The image simply does not extend that far.

The old answers were to crop something else, to mirror a strip of the edge, or to place the photograph on a coloured panel. **Outpainting** extends the canvas and generates plausible content into the new area, conditioned on the existing pixels along the boundary.

Mechanically it is inpainting with the mask outside the original image instead of inside it. All the properties from the first lesson apply: it is softer, cleaner, wrong about text and regular geometry, and non-deterministic.

Technique that works

Extend in stages, not in one jump. Asking for 100 per cent more width in one operation gives the model a thin strip of context along one edge to condition a very large region. It drifts. Extending by 20 to 30 per cent, then again, gives each pass substantial context and a much more coherent result.

Keep overlap. Include a band of the existing image inside the generated region — most tools do this automatically, and where the setting is exposed, 64 to 128 pixels is a reasonable overlap.

Describe the scene, not the extension. "Continues the wooden table and grey wall, soft window light from the left" works. "More table" does not.

Expect the horizon to drift. A horizon or a strong architectural line will bend slightly across the new region. Check it with a straight edge and correct by transforming the generated area or by cloning the real line outward.

Then unify. Grain, vignette, black point and colour across the whole extended image, exactly as in compositing. The generated strip is cleaner than the photograph.

Aspect-ratio conversion, which is the real use

One photograph has to appear as:

- 1:1 for a marketplace grid
- 4:5 for a social feed
- 9:16 for a story or reel
- 16:9 for a slide or a website hero
- 3:1 for a banner

Cropping to all five destroys the composition in at least three of them. Outpainting the background outward, then cropping, keeps the subject where it belongs in each format.

The practical workflow: extend the master file generously in both directions once, keeping the extension on its own layers, then crop that single master to each ratio. One generation, five outputs, and the subject sits correctly in all of them.

This is genuinely the most useful application of generative imaging in everyday commercial work, and it is the least discussed.

Alternatives that need no model

For a modest extension, three older techniques still work and are exact:

Mirror the edge. Copy a strip, flip it, and blend it in with a soft mask. Invisible on texture — grass, water, sky, bokeh — and obvious on anything with recognisable content.

Content-aware scale (*Edit > Content-Aware Scale*, and GIMP's *Liquid Rescale* plugin, which is the free implementation of seam carving). This stretches a photograph while protecting important regions, by removing or duplicating low-energy vertical seams. It works remarkably well on a landscape with a lot of sky and fails visibly on anything with structure.

Extend with a graphic panel. Put the photograph on a coloured or textured field and use the space for type. This is design rather than illusion, it is honest, and for many layouts it looks better than a generated extension.

Consider these first. They are faster, free, deterministic, and they invent nothing.

Free tools

ComfyUI, **AUTOMATIC1111** and **Fooocus** all do outpainting with local Stable Diffusion models. Krita's AI diffusion plugin does it on a canvas you can then keep editing. Several phone apps offer it.

Adobe's Generative Expand is the integrated paid version, available from the crop tool.

The honesty question, which is different here

Outpainting raises a question that inpainting does not, and it is worth stating plainly.

Removing an object is a subtraction that a viewer may or may not notice. **Extending a photograph adds content that the viewer will read as part of the scene** — more of the room, more of the street, more of the landscape. They will believe it was there, because everything else in the frame was.

So:

- **Advertising, book covers, banners, design.** Unremarkable.

- **Editorial and documentary.** Not acceptable. Extending a news photograph fabricates the scene around the event.
- **Property.** An extended room shows more room than exists. This is the sky-replacement problem with a sharper edge to it.
- **Portraits and people.** Extending the frame around a person often generates additional limbs, objects or figures at the boundary. Check.

The delivery note habit applies: "canvas extended by generative fill on left and right; original frame supplied."

The limit

The further you extend, the less the original constrains the result, and the more the output reflects the training data rather than your photograph. A 20 per cent extension of a textured background is usually indistinguishable. A photograph extended to three times its width is mostly a generated image with a photograph in the middle of it, and it will look like one — the generated regions share a smoothness, a colour character and a compositional blandness that becomes obvious at scale.

When the extension exceeds roughly half the original dimension, stop and consider whether the honest answer is a different photograph.

BEFORE YOU MOVE ON

A designer needs one photograph in five aspect ratios. What is the most efficient approach that keeps the subject correctly placed in each?

A. Crop to each ratio from the original, accepting that the subject will sit differently in each, since cropping is exact and introduces nothing.

B. Outpaint separately for each ratio, since each extension can then be tailored to that format's requirements.

C. Extend the master generously in both directions once, keeping the extension on its own layers, then crop that single master to each ratio.

D. Use content-aware scale to stretch the image to each ratio, which protects the subject while adjusting the background.

62 · 9 min

Which upscaler, for which material

THE ONE THING TO KEEP

Upscalers come in families with different tendencies to invent texture, they must be run after denoising and before sharpening, tiling seams are fixed with a larger overlap, and 2× is close to recovery while 8× is mostly the training data rather than your photograph.

Three families, three behaviours

Model-based upscalers are not one thing. Knowing which family you are using explains the result.

GAN-based super-resolution — ESRGAN, **Real-ESRGAN**, and the many community models built on them. A generator produces a larger image and a discriminator judges whether it looks real; the generator learns to produce convincing texture. The result is sharp and detailed and the detail is invented. These are the workhorses, and Real-ESRGAN in particular was trained on synthetically degraded images so it handles real-world compression and blur rather than only clean downscales.

Transformer-based — SwinIR and successors. Generally more faithful and less inclined to hallucinate texture, slower, and often slightly softer.

Diffusion-based — the newest, most capable of producing convincing detail and the most likely to produce detail that has nothing to do with your image.

And separately, **face restoration** models — GFPGAN, CodeFormer — which do not upscale generally but regenerate faces from a learned prior of what faces look like. They are frequently bundled with upscalers and they are the component that changes who somebody is.

Two families of upscaler, and what each one is for

	Texture it invents	Where it belongs
-ESRGAN, a GAN	**A great d[illegible]** trained [illegible] discriminator	**Skin, fabric, foliage, brickwork** 2 times is close to recovery, 4 times is interpretation, 8 times is illustration
IR, a transformer	**Comparatively [illegible]** slower, and often slightly [illegible]	**Material that must not change** and none of it is evidence of anything

Both predict a plausible high-resolution image consistent with weak evidence. Where the evidence is strong the plausible answer is close to the true one; where it is weak the output tells you about the training set. There is no warning at the boundary, and the result looks equally confident on both sides of it.

Picking by material

- **Skin, fabric, foliage, brickwork, general photographic texture.** Real-ESRGAN at 2× is excellent. This is what it was built for.
- **Anime, illustration, flat graphics.** Use a model trained on that material — `realesrgan-x4plus-anime` and equivalents. A photographic model on line art produces mushy edges.
- **Text, logos, signage.** Do not. The letterforms are regenerated and come back the wrong shapes. Rebuild the text as vector.
- **Faces.** Use face restoration only when a plausible face is acceptable. On a family photograph most people are delighted. On anything identifying, it produces a different person, subtly.
- **Heavily compressed sources.** Real-ESRGAN's degradation-aware training handles these better than most. Expect the JPEG blocks to be interpreted as content.
- **Evidence of any kind.** No.

The order that matters

Denoise before upscaling, not after. An upscaler trained on clean-ish inputs will interpret noise as texture and enlarge it into structured artefacts — wormy patterns, clumps, pseudo-detail. Remove the noise first and the model has signal to work from.

Do not sharpen before upscaling. The halos get enlarged too. Sharpen at the end, at the output size, exactly as in the sharpening lesson.

Crop first. Upscaling a 40-megapixel file to use a 600-pixel corner of it is minutes of computation thrown away.

2× twice is not the same as 4× once. Running a 2× model twice compounds its invented detail on top of its own previous inventions, and the second pass has no real information to work with. Use the 4× model if you need 4×.

Tiling, and the seams it causes

A large image does not fit in a graphics card's memory, so upscalers process it in tiles and join them. Where the tiles meet, the model had different context on each side, and a seam can appear — usually a faint line of different texture or brightness.

The fix is a **tile overlap** setting, usually called padding or tile pad. Raising it from the default 10 pixels to 32 or 64 removes most seams at a modest cost in time. In Upscayl and in the command-line `realesrgan-ncnn-vulkan` this is exposed directly.

If you see a regular grid of faint lines in an upscaled image, that is tiling, and it is fixable.

Running it free

Upscayl is a free, open-source desktop application wrapping Real-ESRGAN and several other models, with a plain interface, batch processing and no upload. It runs on Windows, macOS and Linux, uses the GPU through Vulkan where one exists, and works on integrated graphics slowly.

```
realesrgan-ncnn-vulkan -i input.jpg -o output.png -n realesr-
gan-x4plus -s 4 -t 256
```

That is the command-line version, the same engine, scriptable for a batch.

chaiNNer is a free node-based tool for building upscaling pipelines and running community models. The **OpenModelDB** collection hosts hundreds of community-trained models for specific material — old film, scanned prints, compressed video, particular game textures — and one of them is often much better on your specific material than a general model.

On the paid side, Topaz Gigapixel and Adobe's Super Resolution inside Camera Raw are convenient and, on ordinary photographic material, not dramatically better than the free options.

Checking the output

Compare at 100 per cent against a bicubic enlargement of the same file. Look specifically at:

- **Faces** — is this the same person?
- **Text and numbers** — are the characters the same characters?
- **Repeated fine texture** — fabric weave, distant foliage, a wire fence. Is the pattern the pattern that was there, or a convincing substitute?
- **Flat areas** — has the model invented texture in what should be smooth sky or wall?
- **Tile boundaries** — any regular grid of faint lines?

If the upscale fails any of those and the picture depends on that region, use the bicubic version, which is honest about being soft.

The limit, restated because it keeps mattering

An upscaler predicts a plausible high-resolution image consistent with the low-resolution evidence. Where the evidence is strong — a well-exposed, in-focus photograph enlarged 2× — the plausible answer is usually close to the true one and the result is genuinely useful.

Where the evidence is weak — a 200-pixel face enlarged 8× — the model is mostly drawing on its training data, and what comes out tells you about the training set rather than about your photograph. There is no warning at the boundary between those two cases, and the output looks equally confident in both.

The practical rule: **2× is usually recovery-like, 4× is interpretation, 8× is illustration.** Say which one you did.

BEFORE YOU MOVE ON

An editor upscales a noisy ISO 6400 photograph 4× and gets wormy, clumped pseudo-detail across the shadows. What should they have done differently?

A. Run the 2× model twice instead of the 4× model once, so each pass had less to invent.
B. Increased the tile overlap, since the artefacts come from the model having inconsistent context between tiles.
C. Denoised the file first, so the model had signal rather than noise to enlarge.
D. Used a transformer-based model, which does not hallucinate texture and would have left the noise unchanged.

---

63 · 9 min

Reconstruction sold as noise reduction

THE ONE THING TO KEEP

Learned denoisers predict a clean image rather than smoothing a noisy one, which is why they work on raw data whose noise statistics are known, why they deliver two to three stops, and why fine repeated texture and small faces come back plausible rather than correct.

What separates it from a slider

Classical noise reduction analyses the local neighbourhood and smooths what looks random while trying to protect what looks like an edge. It has no idea what the picture contains, so it faces the unavoidable trade from the earlier lesson: luminance noise and fine texture are the same thing at the pixel level.

A learned denoiser was trained on pairs of noisy and clean images of real scenes. It has learned what clean photographs of fabric, foliage, skin and brick look like, so it does not smooth — it **predicts** the clean version. That is why the results are so much better, and why they carry a different risk.

The headline improvement is genuine: on a high-ISO raw file, the good implementations give the equivalent of two to three stops. An ISO 12800 photograph comes back looking like ISO 1600.

Why several of them are raw-only

Adobe's Denoise, DxO's DeepPRIME and several others work on the raw file before demosaicing.

The reason is that noise in raw data has known statistics — it is dominated by photon shot noise, which is Poisson-distributed and therefore predictable from the signal level, plus a read noise component characteristic of the sensor. A model trained on that distribution has a well-defined problem.

Once the file has been demosaiced, white-balanced, tone-curved and JPEG-compressed, the noise has been through several non-linear transformations and is correlated between neighbouring pixels. It no longer matches what the model was trained on. That is why denoising a JPEG is markedly less effective than denoising the raw, and why these tools often refuse JPEGs entirely rather than doing a poor job.

It is also a strong practical argument for shooting raw in low light, separate from the exposure argument.

The characteristic failures

Fine repeated texture is replaced. Fabric weave, distant foliage, hair, a wire fence, small text on a sign. The model produces texture that is statistically right and locally wrong. On a portrait nobody cares; on a photograph of a document it matters entirely.

Waxy skin at high strength. The model's prior for skin is smooth, so pushed hard it returns smooth skin. This is the same plastic look as over-aggressive classical denoising, arrived at differently.

Small faces become different faces. Where a face occupies few pixels, the model fills in from its learned prior of faces. Group photographs are where this shows.

Stars and specular points vanish. They look like impulse noise. Astrophotographers turn these tools off for that reason.

Detail that is not in the training distribution is handled badly — unusual materials, scientific imagery, microscopy, infrared.

Deblurring, which is weaker than it appears

The unrecoverable-losses lesson explained why blind deconvolution is ill-posed. Learned deblurring does not solve that; it sidesteps it by generating a sharp-looking image consistent with the blurred input.

On mild, simple blur it produces genuinely useful results. On anything substantial it produces a confident, sharp, plausible image whose fine detail is invented. The giveaway is usually over-clean edges with a slight plasticity, and text that reads as text from a distance and not close up.

Treat any "unblur" or "enhance" feature as generation, and apply the same test as everywhere else: does the content of this region matter as a record?

Free tools

- **darktable's profiled denoise** uses measured noise profiles for specific camera and ISO combinations, so it knows the statistics of your sensor's noise at that setting. It is not a learned model and on many files it is competitive with the paid learned ones, particularly its wavelet and non-local means modes used together — chroma with non-local means, luminance with wavelets.
- **RawTherapee's** noise reduction is also strong and highly configurable.
- **Neat Image** and **DxO** are the paid standards.

- **G'MIC** inside GIMP has a large set of denoisers including some learned ones.
- Several Real-ESRGAN variants denoise as a side effect of upscaling at 1×, which is a usable free trick: upscale 2× with denoising and downscale back.

darktable's approach deserves particular mention because it is the honest one: it uses a measured profile of your actual sensor rather than a prior about what photographs look like, so it removes noise without inventing content. It is less dramatic and more truthful.

How to use a learned denoiser well

1. **Apply it to the raw**, early, before anything else.
2. **Use less than the default.** Most implementations default high. Half strength usually keeps the texture and removes the objectionable noise.
3. **Compare at 100 per cent against the undenoised version**, looking specifically at fabric, hair and foliage.
4. **Mask it off areas that must stay authentic**, where the tool allows. Some do not, in which case process twice and blend layers by hand.
5. **Add grain back** if the result looks synthetically clean next to the rest of a set.
6. **Sharpen after**, never before.

The statement to hold

A learned denoiser is a reconstruction tool sold as a cleaning tool. For a wedding, a portrait, a concert, a family photograph — almost all paid photographic work — it is a large improvement and the reconstruction is invisible and harmless.

For a photograph that will be used as evidence of what something looked like — a document, an identifying detail, a scientific measurement, a news image, an insurance claim — the output is not a cleaned version of the original. It is a new image consistent with the original, and it should be labelled as processed or not used at all.

That distinction is not a slider setting. It is a decision about what the picture is for, and it is made before you press the button.

BEFORE YOU MOVE ON

A photographer finds that a learned denoiser produces a dramatic improvement on their raw files but barely helps on the JPEGs from the same camera. What explains the difference?

A. JPEG compression has already removed most of the noise, so there is less for the denoiser to work on.

B. The model was trained on raw sensor noise, whose statistics are predictable, and demosaicing, white balance, tone curves and compression transform that noise into something it was not trained on.

C. JPEGs are 8-bit, so there is not enough precision for the model to distinguish noise from signal.

D. Denoisers require the demosaic step to be performed by their own algorithm, and a JPEG has already been demosaiced by the camera.

64 · 8 min

A guess about what colour the dress was

THE ONE THING TO KEEP

Colourisation predicts chrominance from luminance using statistical associations learned from modern photographs, so it is reliable on sky, foliage and skin and a hedge towards muddy averages on anything manufactured — and none of it is recovered, because the greyscale file holds no colour information at all.

What the model is predicting

A colourisation model takes the luminance channel — the greyscale photograph — and predicts the two chrominance channels that would go with it. Working in a space like Lab, where lightness is separate from colour, means the original tones are preserved exactly and only colour is added.

It was trained by taking millions of colour photographs, discarding their colour, and learning to predict it back. So it has learned strong statistical associations: sky is blue, grass is green, skin is in a particular range, wood is brown, water reflects sky.

Those associations are exactly as reliable as the association is in the world.

What it gets right and wrong, predictably

Reliable: sky, foliage, grass, water, skin in the range the training data covered, wood, brick, sand, stone. These have narrow colour distributions in reality, so the prediction is usually close.

Unreliable: clothing, painted objects, vehicles, signage, flowers, furniture, and anything manufactured. A dress in a 1940s photograph could be any colour. The model produces a plausible one — often a desaturated blue, brown or muted red, because those are the average of the possibilities, and averages are what a model trained with a squared-error objective produces.

That last point explains the characteristic look. Colourised photographs are frequently **undersaturated and slightly brown**, because when the model is uncertain it hedges towards the mean of the plausible colours, and the mean of all possible dress colours is a muddy neutral. Models trained with adversarial objectives commit more strongly and are more often confidently wrong.

Systematically wrong in one direction you should know about: skin tone prediction inherits the composition of the training data, which as the skin tones lesson described has historically under-represented darker skin. Colourisation of photographs of people whose skin sits outside the well-represented range is measurably less reliable.

Free tools

DeOldify is the best-known open-source colouriser, available as a notebook you can run in Google Colab for free, or locally. It has two models: an "artistic" one that commits to more vivid colour and makes more mistakes, and a "stable" one that is more conservative. It also does video.

There are newer open models, and several image editors and phone apps have a colourise button that wraps one of them.

GIMP has a G'MIC colourise filter, and separately supports the hand method below.

Hand colourising, which is often better

The technique predates the models by a century and it produces a result you can defend.

1. Keep the greyscale photograph as the base layer.
2. Add a new layer set to **Colour** blend mode above it.
3. Paint flat colour onto that layer with a soft brush. Because the blend mode takes hue and saturation from your paint and luminosity from the photograph, all the tone and texture of the original shows through.
4. Use a separate layer per region — skin, clothing, background, hair — so each can be adjusted independently.

5. Reduce each layer's opacity to 40 to 70 per cent. Fully saturated hand colouring looks like a postcard.

The great advantage: **you decide, and you can be told.** A family can say the coat was navy, the car was green, the house was whitewashed. That information exists in people's memory and in written records, and a model has no access to it. A hand-coloured photograph informed by the family is a reconstruction based on evidence; a model-coloured one is a reconstruction based on statistics.

A sensible hybrid: run the model to get a base, then correct the regions where somebody actually knows, using Colour-mode layers on top.

The honesty question, which is unusually sharp here

Colourisation is popular, moving and widely shared, and it is also the operation most likely to be mistaken for a recovery. People see a colourised 1912 photograph and experience it as seeing the past in colour. What they are seeing is a greyscale photograph with a model's guess painted over it.

This matters most for historical and documentary material. Historians have raised consistent objections: colourisation removes the viewer's awareness that they are looking at a mediated document, it asserts specific facts — uniform colours, flag colours, skin tones — that may be wrong, and the confidence of the result discourages checking.

The positions people take:

- **For family photographs**, where the audience knows what it is and the emotional value is the point, most people consider it entirely fine.
- **For museum, archive and educational use**, the widespread convention is to present the original alongside, label the colourisation clearly, and document what the colours were based on.
- **For anything asserting historical fact** — a uniform's colour, a person's appearance, a flag — a guess should not be presented as a finding.

The practical habit, as elsewhere in this course: deliver both files, label the colourised one, and write one line about where the colours came from.

The limit

The photograph contains no colour information. None. It was never recorded, and there is no signal to recover — this is not like a faded colour print where a weak dye layer survives.

Everything the colouriser produces comes from outside the photograph, either from a model's statistics or from your knowledge. That is the entire content of the operation, and it is why the honest description of the output is a colour illustration derived from a photograph rather than a colour photograph.

BEFORE YOU MOVE ON

A colourised 1930s street photograph has convincing sky, foliage and skin, but every coat and every shopfront is a similar muted brown-blue. What does this pattern indicate?

A. The model was run at too low a resolution, so small objects received an averaged colour from their surroundings.
B. The photograph's emulsion preserved more information in natural subjects than in dyed fabrics, so less signal was available for the manufactured objects.
C. Natural subjects have narrow real-world colour distributions, while manufactured objects could be any colour, so the model hedges towards the average of the possibilities.
D. The stable DeOldify model was used rather than the artistic one, and switching models would produce accurate garment colours.

65 · 9 min

On your own machine, with no graphics card

THE ONE THING TO KEEP

Background removal, upscaling and denoising run usefully on a laptop with no graphics card, inpainting runs slowly but workably, and running locally buys confidentiality, zero per-image cost and independence from a service — while Colab's free GPU is a real option that is not local.

Why local matters beyond cost

Three reasons to run models yourself rather than uploading.

Confidentiality. A client's unreleased product, a photograph of a child, a medical image, anything under an agreement. Uploading it to a service means it has left your control under terms you probably have not read, and several services reserve rights to use uploads for improvement.

Cost at volume. Four hundred background removals or upscales through a paid service is a line on an invoice. Locally it is electricity.

Availability. A service that shuts down, changes its terms or raises its prices takes your workflow with it. A model on your disk does not.

The hardware reality, stated plainly

With a discrete GPU (NVIDIA with 8 GB or more of video memory is the comfortable case): everything in this module runs at a usable speed. An inpaint is seconds; an upscale is seconds.

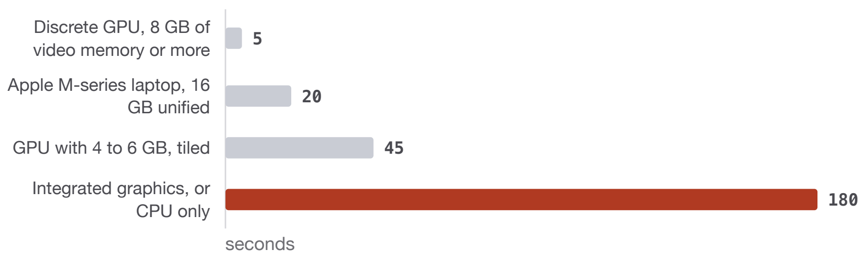
With 4 to 6 GB of video memory: everything still runs, with smaller batch sizes and tiling. An inpaint is tens of seconds.

With integrated graphics or on a CPU only: background removal is two to five seconds and completely practical. Upscaling a 12-megapixel image is one to several minutes and practical for a handful. Diffusion inpainting is one to five minutes per generation, which is slow but workable when you need three fills rather than three hundred.

On Apple Silicon: the unified memory architecture works well and most tools support the Metal backend. An M-series laptop with 16 GB is a genuinely capable machine for this.

On a phone: some on-device background removal and upscaling exists; diffusion generally does not, and apps that offer it are usually uploading.

One inpaint generation, on four machines



Background removal is two to five seconds on every one of these, which is why rembg on an ordinary laptop is a complete answer for a batch of cut-outs. Diffusion is where the hardware shows, and slow is not the same as impossible: three fills at three minutes each is a workable afternoon, three hundred is not.

The free stack, by job

Background removal — `rembg`, installed with `pip install rembg`. CPU-friendly, batch-capable, several models.

Upscaling — Upscayl, a desktop app with a plain interface and no upload. Or `realesrgan-ncnn-vulkan` from the command line, which uses Vulkan so it works on AMD, Intel and NVIDIA alike.

Inpainting and generation — Fooocus is the gentlest entry point: one download, sensible defaults, an inpaint mode. **AUTOMATIC1111's web UI** is the established general tool. **ComfyUI** is a node graph, more capable and steeper. **Krita's AI diffusion plugin** puts local inpainting directly on a Krita layer with masks, which for an image editor is the most useful integration available anywhere, free or paid.

Colourising — DeOldify, locally or in a free Google Colab notebook.

Denoising — darktable's profiled denoise, which is not a learned model and is excellent.

Everything else — chaiNNer builds pipelines from community models; **OpenModelDB** hosts them; **Hugging Face** hosts the model weights for most of the above along with model cards describing training data and licences.

Google Colab, and what it is for

Colab gives you a Python notebook running on Google's hardware, with a free tier that includes a GPU for limited sessions. For someone with no graphics card it converts an impossible task into a slow but workable one.

The honest limits: the free tier's GPU availability varies and can be refused at busy times, sessions disconnect after a period of inactivity and everything not saved is lost, and **whatever you upload goes to Google's machines**, which puts it back in the confidentiality category. For personal or public images it is a genuine free route; for client-confidential material it is not local, whatever the notebook is called.

Kaggle offers a similar free notebook environment with GPU hours.

Licences: the part people skip

Models have licences, and they differ substantially.

- Some are permissively licensed for any use including commercial.
- Some carry **use restrictions** — the RAIL family of licences, used by several well-known image models, permits commercial use while prohibiting specified applications.
- Some are explicitly **research-only**, and using them for client work breaches the terms.
- Some community fine-tunes have unclear provenance and inherit whatever their base model required, which is often unstated.

The model card on Hugging Face states the licence. Reading it takes two minutes and is the difference between a defensible workflow and one you cannot describe to a client.

This is separate from the copyright status of the *training data*, which is genuinely unsettled and is the subject of the last lesson in this module.

A practical starting setup

For an editor on an ordinary laptop who wants to stop uploading things:

1. `pip install rembg` — background removal today.
2. **Upscayl** — upscaling today.
3. **darktable** — raw development and denoise today.
4. **Fooocus** or **Krita's AI diffusion plugin** — inpainting when you have a spare afternoon to set it up.

That is four installations, no subscription, nothing uploaded, and it covers the large majority of what generative tools are used for in image editing.

The trade you are making

Local models are usually a step behind the best hosted ones. Adobe's generative fill is more convenient and often produces a better first result than a local Stable Diffusion inpaint. Paid upscalers have better interfaces.

What you get in exchange is control: nothing leaves the machine, nothing depends on a service, nothing costs per image, and you can read what the model was trained on.

For a hobbyist the convenience may well win. For anyone handling other people's photographs professionally, the ability to say truthfully that a client's images never left your computer is worth more than a slightly better first attempt, and it is increasingly something clients ask about.

BEFORE YOU MOVE ON

A retoucher under a non-disclosure agreement needs to inpaint an unreleased product photograph and has no graphics card. Which option satisfies the agreement?

A. Use a free Google Colab notebook with a GPU, since it is a free and self-run environment rather than a commercial service.
B. Run a local diffusion inpainting tool on the CPU, accepting one to five minutes per generation.
C. Use a web-based background removal and inpainting service, but delete the uploaded file afterwards.
D. Crop the image to hide identifying detail, then use any hosted tool on the remainder.

66 · 8 min

What can be proved about where an image came from

THE ONE THING TO KEEP

C2PA signs a chain of edits rather than certifying truth, invisible watermarks live in the pixels but only for vendors that apply them, and AI-detection services are too unreliable to accuse or defend anyone with — so the strongest provenance a working editor has is the kept original plus a written delivery note.

Three different mechanisms, often confused

Metadata provenance — a record attached to the file saying what was done to it. C2PA and Content Credentials are this.

Invisible watermarking — a signal embedded in the pixels themselves, designed to survive editing and re-encoding. Google's SynthID is this.

Detection — analysing an image and guessing whether it was generated. This is the least reliable of the three.

They answer different questions and have different failure modes.

C2PA and Content Credentials

The Coalition for Content Provenance and Authenticity publishes an open specification for a **cryptographically signed manifest** attached to an image. The manifest records the capture device or software, what edits were applied and by which tool, and whether generative tools were used. Each step signs the previous, producing a chain.

What it genuinely provides: if the chain is intact and the signatures verify against a trusted issuer, you have a strong record of what the file went through. Some cameras now sign at capture. Adobe's applications write Content Credentials. Several generative services attach a manifest stating the image was AI-generated.

What it does not provide:

- **It proves the chain, not the truth.** A signed manifest on a photograph of a staged scene certifies that the camera recorded it, not that anything in front of the camera was real.
- **Absence proves nothing.** Almost no images have credentials. A file without a manifest is the overwhelming default, not a suspicious anomaly.
- **Metadata is trivially removed.** Stripping it is one command. It is not tamper-*proof*, it is tamper-*evident*: you can tell the chain was broken, and you cannot stop it being broken.
- **Platforms strip it.** Most social platforms re-encode uploads and discard metadata, which removes credentials along with EXIF. Some now preserve or display them; most do not.
- **A screenshot destroys it entirely**, and a screenshot is indistinguishable from a photograph of a screen.

The standard is worth supporting and it is not an authenticity guarantee. Treat a valid manifest as evidence, an absent one as no evidence either way, and a broken one as a reason to ask questions.

Invisible watermarking

A pixel-level signal, imperceptible to a viewer, that a matching detector can recover. Google's SynthID is applied to images generated by several of its models and is designed to survive cropping, re-encoding, filtering and moderate resizing.

Stronger than metadata, because it is in the image rather than attached to it. Still not absolute: heavy editing, large crops, severe compression, screenshotting through a display, and deliberate adversarial processing all degrade or remove the signal, and detection requires the corresponding detector, which is not universally available.

And it is per-vendor. An image from a model that does not watermark carries no mark, which again means absence proves nothing.

Detection tools

Services claiming to tell you whether an image is AI-generated are the weakest of the three, and it is important to be blunt about this because they are widely cited.

They work by learning artefacts of particular generators, so they degrade as generators change, they produce false positives on heavily processed real photographs, and they produce false negatives on generated images that have been edited or re-encoded. Published evaluations repeatedly find accuracy well below what the marketing claims, and the error rates are not uniform across image types.

A detector's output is a weak signal, not a finding. Do not accuse anyone on the strength of one, and do not defend yourself with one.

Visible watermarks, for a different purpose

These are about attribution and unauthorised use, not provenance.

- A visible mark deters casual reuse and is removable with the tools in this course, increasingly easily.
- Place it **over the subject**, not in a corner, if deterrence is the point — a corner mark crops off.
- Low opacity over a busy area is harder to remove than a solid mark on a plain background.
- For client proofs, a large diagonal mark across the whole frame is normal and expected.

The honest assessment: a visible watermark stops opportunistic reuse and does not stop anyone determined. Its main practical value for a working photographer is attribution — people who see the image know whose it is.

What a working editor should actually do

- **Keep your originals.** The single most useful provenance you can have is the untouched camera file with its EXIF, stored somewhere you can find it. In any dispute about what you did, that plus your layered working file is far stronger evidence than any standard.
- **Write a delivery note** saying what was done. One line. This has come up in every module of this course because it is the cheapest form of provenance and the most widely useful.
- **Fill IPTC creator and copyright fields** on delivered files. They will often be stripped, and they cost nothing and sometimes survive.
- **Enable Content Credentials** where your tools support it. It costs nothing and it contributes to a standard that becomes useful as adoption grows.
- **Do not rely on any of this to protect you.** The protection is the contract, the record and the relationship.

The unsettled part

Whether any of this should be mandatory, and in what form, is being legislated differently in different places and the requirements are moving. Several jurisdictions have introduced or proposed obligations to label synthetically generated content; the definitions, the thresholds and the penalties differ, and some are being challenged.

There is no single rule to give you, and anyone who states one confidently is describing one jurisdiction. What is stable enough to act on: keep originals, disclose generative work in writing, and do not represent generated content as a record of something that happened. Those satisfy the spirit of every rule proposed so far, and they were good practice before any of them existed.

BEFORE YOU MOVE ON

A client receives a photograph with no Content Credentials attached and asks whether this means it was manipulated. What is the correct answer?

A. Yes, since credentials are written automatically by editing software and their absence indicates they were deliberately stripped.

B. No, because almost no images carry credentials and platforms routinely strip metadata, so absence is the default and carries no information.

C. It cannot be determined without running an AI-detection service on the file, which would settle the question.

D. No, because credentials only record generative edits, so an image edited conventionally would never carry one anyway.

67 · 9 min

The decision, and the parts nobody has settled

THE ONE THING TO KEEP

Ask whether the picture is a record or a construction, because that determines whether generated content is permissible — and treat the copyright status of model training as genuinely unresolved, documenting what you used rather than assuming an answer.

One question, asked before the tool is opened

> ***Is this picture a record of something, or a construction?***

A record asserts that the world looked like this. A construction asserts nothing beyond itself. Almost every difficult case in this module resolves once you have answered that, and the answer depends on the picture's use rather than on how it was made.

The same photograph of a shop can be a record — in a news story about that shop, in an insurance claim, in a planning application — and a construction, on a poster advertising it. The file is identical. The obligations are not.

Where generated content is not acceptable

News and documentary. The major wire services and photojournalism competitions prohibit adding or removing content. Only what a darkroom could do — exposure, contrast, colour, crop — is permitted. Photographers have lost their careers over a single cloned sensor spot. Generative fill is not in a different category from cloning here; it is a more capable version of the same prohibited act.

Evidence. Insurance, legal, forensic, accident, inspection. A denoised, upscaled or filled region is not a record of what was there. If the image will be relied on, process it minimally and keep the original.

Medical and scientific imaging. Learned denoising and upscaling produce plausible structure. In an image used for diagnosis or measurement, plausible structure is a hazard, and the relevant professional standards address this directly.

Identification. Faces, number plates, documents, serial numbers. Face restoration models regenerate features. An upscaled number plate is a guess.

Products, where the buyer receives the item. Covered in the retouching block, and generative tools make it easier to cross the line without noticing.

People's likeness without consent. Generating a person doing something they did not do, placing a real person in a scene, altering someone's body or face beyond what they agreed to. This runs into personality and publicity rights, defamation, and in many places specific new offences around synthetic intimate imagery. It is also simply not something to do.

Where it is uncontroversial

Advertising, illustration, book covers, posters, concept work, game art, mood boards, backgrounds for design, extending a canvas for a layout, restoring a family photograph for a frame, removing a distraction from a personal picture.

In these, nobody is being told that the world looked like this, and the question does not arise.

The middle, where judgement lives

Real estate. Widely done, increasingly complained about, disclosure conventions emerging and varying by market.

Portraits and weddings. Retouching is expected. The line is the subject's expectations, which is why asking is the answer.

Editorial illustration — an image accompanying an opinion piece rather than a report. Usually acceptable with a label.

Corporate and internal. Usually fine, occasionally not, and a line in the delivery note covers it.

Competitions. Rules vary enormously and change. Read them, for the specific category, this year.

The working habit that covers all of these: **write down what you did.** Every module of this course has come back to that, because it converts an unstated assumption into a stated one, and almost every dispute in this trade is an unstated assumption meeting a different unstated assumption.

Labelling requirements, as they actually stand

Several jurisdictions have introduced or proposed obligations to disclose synthetically generated or substantially altered content. The scope, the definitions, the thresholds and the enforcement differ between them; some target platforms, some target creators, some target specific categories such as political advertising or intimate imagery. Some are in force, some are phasing in, and some are being challenged.

There is no single rule to state, and a course that gave you one would be describing one country and misleading everybody else. What is generally safe:

- Disclose generative work in writing to your client.
- Do not present generated content as a record of an event.
- Check the specific rules for your country and your sector, this year, and for anything with legal exposure ask a lawyer in your jurisdiction. Nothing in this course is legal advice.

The genuinely unresolved question

Whether training image models on copyrighted works without permission is lawful is **not settled**, and it is important to present the shape of the disagreement rather than a conclusion.

One position: training is a transformative analytical use that does not reproduce the works, comparable to a person learning from looking, and treating it as infringement would make machine learning on public material impossible.

The other: the works were copied to build a commercial product, the creators were neither asked nor paid, and the output competes directly with the market for the originals.

Between them sit questions courts are actively working through: whether the copying for training is fair use or fair dealing, whether an output that resembles a training work infringes, whether a model's weights are a derivative work, whether text and data mining exceptions cover commercial training, and what opt-outs mean in practice. Different jurisdictions have reached different provisional answers, litigation is ongoing in several, and legislation is moving.

There is also an **output-side** question that is separate and less discussed: whether an image produced by a model can be copyrighted at all. Several copyright offices have taken the position that purely machine-generated output lacks the human authorship their law requires, with protection possible for human-authored arrangement and modification. If a client needs to own what you deliver, that matters commercially, and it is worth raising before the work rather than after.

What to do while it is unresolved

- **Read the model's licence**, which is a contract question and is settled, unlike the copyright question.
- **Prefer models with documented training data** where you have a choice.
- **Tell clients when generative tools were used**, so the ownership question is theirs to weigh.
- **Do not reproduce a living artist's identifiable style on commission** and represent it as generic. Whatever the copyright position turns out to be, that is a reputational and possibly a passing-off problem now.
- **Keep your own originals**, which is the one thing that remains valuable under every possible outcome.

The honest summary: the tools are useful, several of their legal foundations are contested, and the contest is not close to finished. Working with that uncertainty means documenting what you did and being straightforward about it, which is a smaller burden than it sounds and is the same discipline this course has asked for from the first module.

BEFORE YOU MOVE ON

A photographer is asked to use generative fill to remove a distracting parked van from a photograph. Which consideration determines whether it is acceptable?

A. Whether the result is convincing enough that no viewer would detect the edit.

B. Whether a conventional tool such as content-aware fill could have achieved the same removal, since only generative tools raise the question.

C. What the picture is for — whether it asserts that the scene looked like this, or is a construction where no such claim is made.

D. Whether the client has authorised the removal in writing, since their instruction settles the question of acceptability.

CASE STUDY · MODULE 7

One classroom photograph, five shapes, and a consent form that did not mention it

An education NGO in Ranchi, preparing a donor report, a website hero and three social formats from a single good photograph.

One frame that works

Out of four hundred photographs from a field visit, one is right: light falling through a doorway onto a teacher at a board, nine children at desks, a wall of exercise books behind. It is a 4:3 frame with the teacher a third of the way in from the left.

The report needed it at five shapes. Square for the donor's grid, 4:5 for the feed, 9:16 for a story, 16:9 for the website hero, and 3:1 for a banner across the top of a landing page. Cropping to all five works for two of them and destroys the composition in three — the 3:1 crop leaves a strip containing the teacher's shoulder and half a child.

The obvious answer is generative expand: extend the canvas and let a diffusion model fill the new area, conditioned on the pixels along the boundary. It is genuinely the most useful everyday application of generative imaging in commercial work, and it is the least discussed.

What made it not obvious here

The photograph shows nine identifiable children. Their parents had signed a consent form, written in Hindi and English, covering the NGO's annual report and its website.

The form says nothing about uploading the photograph to a third-party service. A hosted generative expand means sending an identifiable image of minors to a company whose terms of service nobody in the office had read, several of which reserve rights over uploads.

The communications officer laid out three options in a note to the programme head.

Hosted expand. Two minutes per ratio, the best first result, and it puts the photograph somewhere the consent form does not reach.

Local expand. Fooocus on her own laptop, which has no graphics card. An afternoon to install and understand, then about three minutes per generation. Nothing leaves the machine. It still generates content that was never in that room, and in the first attempts at 9:16 it twice produced an extra child-shaped figure at the lower edge, which is a characteristic failure at the boundary of an outpaint around people.

No model at all. Mirror the edge where the extension is pure texture — floor, wall — and place the photograph on a coloured panel with the report's title for the two extreme ratios. Exact, free, deterministic, invents nothing, and it reads as design rather than as more room.

The cost on the third option is real and worth stating: it looks like a layout rather than like a photograph, and the donor's own materials are full of full-bleed photographs. Choosing it is choosing to look different from everybody else in the folder.

The workflow that would have applied either way

One thing was settled before the confidentiality question arose, because it is settled in every job of this shape: you do not run five separate extensions for five ratios.

Extending a canvas five times from the same original produces five different inventions of the same wall, and they do not agree with each other, so the set looks like five photographs of five rooms. The professional route is to extend the master generously in both directions once, keep the extension on its own layers, and then crop that single file to each ratio. One generation, five outputs, and the subject sits where it should in all of them.

The extension itself has to be staged. Asking for a hundred per cent more width in one operation gives the model a thin strip of real pixels along one edge to condition a very large region, and it drifts — the wall changes colour, the floor line bends. Twenty to thirty per cent at a time, with a generous band of the existing image inside each generated region, gives each pass enough context to stay coherent.

And whatever comes back is cleaner than the photograph it is attached to, so grain, vignette and black point have to be unified across the whole extended frame afterwards.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

The programme head chose the third option for 3:1 and 9:16, with the report's title set on a deep green panel beside the photograph. For 16:9 they used a twenty per cent local outpaint, where the new area was floor and wall texture and the boundary was nowhere near a person, then matched grain and black point across the whole frame so the extended strip did not read as cleaner than the photograph. Square and 4:5 were straight crops.

Nothing was uploaded. The asset folder carries a one-line note against the 16:9 file saying the canvas was extended by twenty per cent on the right by a local model, and the original frame sits beside it.

The part that outlasted the report was the consent form. It was rewritten for the following year, in Hindi and English, to say what the photographs would be used for, that they would not be altered beyond cropping and colour correction without asking again, and that they would not be sent to services outside the organisation. The field team reported that it took about ninety seconds longer per family to explain and that no family refused.

Why is extending a photograph a sharper honesty problem than removing something from one?

Removing an object is a subtraction the viewer may or may not notice. Extending the frame adds content the viewer reads as part of the scene — more of the room, more of the street — and they believe it was there because everything else in the frame was. The addition is not presented as an addition; it is presented as the rest of the photograph. That is why outpainting is unremarkable in advertising and design, where nobody is being told the world looked like this, and not acceptable in editorial or documentary work, where extending the frame fabricates the scene around the event.

Mechanically, why did the outpaint produce an extra child-shaped figure, and what reduces it?

Inpainting and outpainting generate content consistent with the surrounding pixels, and the surroundings here contain a repeating arrangement of seated children. The model has no representation of how many children were in the room; it produces what is statistically likely given the context, and another figure is likely. The same mechanism explains why brick courses drift and why text comes out as letter-shaped marks. What reduces it is extending in smaller stages with generous overlap so each pass is more strongly constrained, keeping the boundary away from people where the composition allows, generating three variations and compositing only the parts that worked onto a masked layer — and checking, every time, rather than accepting a whole result.

What did the local route buy, and what did it cost?

It bought confidentiality — a statement the NGO can make to parents and to a donor that no photograph of an identifiable child left the office — along with zero cost per image and independence from a service that could change its terms. It cost an afternoon of setup, about three minutes per generation on a laptop with no graphics card, and a first result that is usually a step behind the best hosted tools. For a handful of extensions that trade is easy; for three hundred it would not be. The deciding factor was not quality, and it was not money: it was that the consent form did not stretch that far.

MODULE 7 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 You are removing a bin from a lawn with generative fill. Why does an empty prompt usually beat typing the word grass?

A. An empty prompt uses a lower denoising strength by default
B. The surroundings constrain the result far more than words
C. The model refuses single-word prompts and falls back to context
D. Typing a word triggers a policy check that rewrites the request

2 A generative fill leaves a visible seam exactly on the outline of the object you removed. What is the likeliest cause?

A. The document is 16-bit and the model works in 8-bit
B. The denoising strength was set too low for that region
C. The mask was drawn tight to the object
D. The surrounding area was too textured for the model to match it

3 You have a noisy high-ISO frame that needs enlarging and sharpening. What is the correct order?

A. Denoise, then upscale, then sharpen
B. Sharpen first, then upscale, then denoise
C. Upscale first, then denoise, then sharpen
D. Denoise, then sharpen at full size, then upscale

4 A colouriser returns a 1940s photograph with a convincing blue sky and a muddy brown dress. Why the difference?

A. The sky faded less than the dress in the original print
B. Blue is the easiest hue for any model to reproduce accurately
C. The model works in Lab, and skies fall inside its gamut
D. Sky has a narrow real distribution and clothing does not

5 What does running rembg and Upscayl on your own machine buy you, beyond avoiding a per-image fee?

A. Results better than any hosted service
B. A statement to a client that nothing was uploaded
C. Freedom from the model's licence conditions
D. Automatic compliance with disclosure rules on generated content

6 An image arrives with no Content Credentials attached. What does that tell you?

A. Nothing, because almost no images carry them
B. That it was probably generated by a model without provenance
C. That the metadata was deliberately stripped by somebody
D. That it has been through at least one social platform's re-encoder

MODULE 8

Out of the editor: size, sharpness and the delivered file

The edit is finished and nobody has seen it yet. Everything between the layered master and the file somebody opens is its own craft, and it is where careful work is most often thrown away: the resize that aliases a striped shirt into moiré, the sharpening tuned for a size the picture will never be seen at, the wide-gamut profile a platform strips on upload. This block takes the picture out — the arithmetic of how many pixels a destination needs, the resampling kernel doing the work, output sharpening, colour conversion, codecs newer than JPEG, motion, metadata, and what happens to your file after you press upload.

> **BY THE END OF THIS MODULE YOU CAN**
>
> Take a finished edit to a delivered file for any named destination — computing the pixel dimensions from the destination rather than the source, choosing a resampling filter and a colour space by mechanism, sharpening last and at output size, selecting the codec and quality by measurement rather than habit, and predicting what a platform's own re-encoding pipeline will do to the result

Nearest, bicubic, Lanczos: what the resize dialogue is computing

THE ONE THING TO KEEP

A resize invents pixel values between the ones you have: point sampling aliases fine detail into moiré, Lanczos is sharp but rings at hard edges, and averaging gamma-encoded numbers rather than light makes downscaled bright detail on dark backgrounds come out roughly half as bright as it should be.

A resize is a question about pixels that do not exist

Reduce a 4000-pixel image to 1600 and you are asking for a new grid whose samples fall between the old ones. Nothing in the file sits at the new positions. The algorithm decides what each new pixel should be, and the different algorithms are different answers to that question.

The dropdown in the resize dialogue is not decoration. On a photograph the difference is small; on a striped shirt, a wire fence, a mask or a screenshot it decides whether the result is usable.

The five you will meet

- **Nearest neighbour** copies the value of the closest old pixel. No averaging, no new values. It produces hard blocks when enlarging and drops information brutally when reducing. It is correct for exactly two jobs: pixel art, where the blocks are the point, and index maps — a mask where 1 means *sleeve* and 2 means *collar* and an averaged value of 1.5 would be meaningless.
- **Bilinear** takes the weighted average of the four surrounding pixels. Cheap, soft, and never produces a value outside the range it started from.
- **Bicubic** uses sixteen pixels and fits a cubic curve through them. The curve overshoots slightly at an edge, which reads as mild sharpening.

Photoshop's *Bicubic Smoother* and *Bicubic Sharper* are the same algorithm with the overshoot dialled down for enlargement and up for reduction.

- **Lanczos** is a windowed sinc function, usually over 36 or 64 samples. It is the sharpest of the classical filters and it rings: a bright rim on the light side of a hard edge and a dark rim on the dark side, exactly the shape of an over-sharpening halo. On photographs this is usually welcome. On a logo against flat colour it is a defect.
- **Model-based enlargement** — Photoshop's *Preserve Details 2.0*, the upscalers in module seven — does not interpolate. It predicts. That is a different operation with a different honesty problem, and it is covered there.

Reducing is where the damage happens

Most people worry about enlarging. Reducing is where images are quietly ruined.

A photograph of a striped shirt contains detail finer than the grid you are reducing to. If the algorithm simply samples the nearest pixel, those stripes land on the new grid at a beat frequency and you get **moiré** — broad wavy bands of colour that were never in the scene. The same thing produces the shimmering brickwork in a downscaled architectural shot.

The fix is a low-pass filter: average a wide enough neighbourhood that detail too fine to survive is removed before it can alias. That is what a proper resize filter does and what point sampling does not. In ImageMagick the distinction is explicit:

```
magick in.jpg -sample 1600x out-bad.jpg    # point sampling, will alias
magick in.jpg -filter Lanczos -resize 1600x out-good.jpg
```

`-thumbnail` is `-resize` with the metadata stripped and a fast pre-reduction for very large files; it is fine for contact sheets and slightly less careful for delivery.

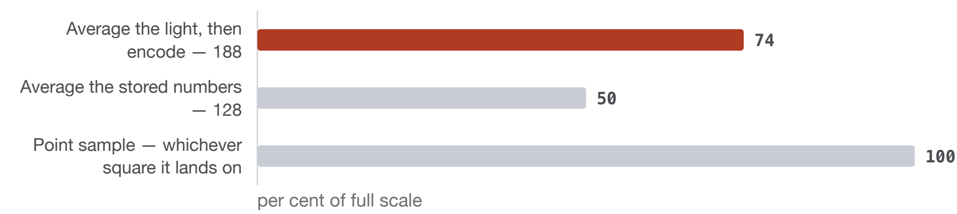
The gamma mistake almost every tool makes

This one is worth the lesson on its own, because it is invisible until you know the pattern, and then you see it constantly.

The numbers in an sRGB file are gamma-encoded — roughly the 1/2.2 power of the light they represent, for the reasons in the gamma lesson. Averaging encoded numbers is not averaging light.

Take a fine black-and-white checkerboard, half the pixels at 0 and half at 255, and reduce it so that each new pixel covers one black and one white. Average the encoded values and you get 128, which displays as about 22 per cent of the light. Average the actual light and you get 50 per cent, which encodes to about 188. The correct result is more than twice as bright as the one most software gives you.

A checkerboard of 0 and 255, reduced until one pixel covers two squares



Half the light is gone, so the honest answer is about 188. Averaging the stored numbers gives 128, which emits roughly 22 per cent of the light — less than half of what it should. Point sampling does not average at all, which is where the moiré on a striped shirt comes from. GIMP and Krita reduce in linear light by default; Photoshop and every web browser do not.

So downscaled images with fine bright detail on dark backgrounds come out too dark: starfields, white text on black, city lights, backlit foliage, a lace edge against shadow. The image does not look wrong in isolation — it looks like the photograph was underexposed.

Resizing in linear light fixes it:

```
magick in.png -colorspace RGB -filter Lanczos -resize 800x -colorspace sRGB out.png
```

`-colorspace RGB` in ImageMagick means linear RGB, which is a confusing name for the right thing. GIMP 2.10 and later resize in linear light by default.

Krita does. Photoshop does not, and neither do web browsers when they scale an oversized image to fit a layout.

The honest qualification: linear downscaling is measurably closer to the physics, and some editors dislike it on skin and foliage because the encoded-space result is slightly contrastier in a way people are used to. It is not a universal improvement in taste. It is a large improvement wherever fine bright detail sits on a dark field.

Two things that are not true

"Resize in 10 per cent steps for quality." This was real advice for the bicubic implementation in Photoshop 6 in 2001. With modern Lanczos or the current bicubic, repeated small steps compound interpolation error and give a softer result than one pass. Measure it yourself if you doubt it: one image, two routes, compare at 100 per cent.

"Always use Lanczos." Its ringing is a genuine artefact. On flat graphics, screenshots and logos it draws grey rims around everything. Use a smoother filter, or better, re-export the graphic from its source at the size you need.

What to choose

- Photographs, any direction: **Lanczos** or bicubic. Check hard edges for rings.
- Screenshots, diagrams, logos: re-export from source. If you cannot, **bilinear** or Mitchell, never Lanczos.
- Pixel art and ID or index masks: **nearest neighbour**, and only at whole-number multiples for pixel art.
- Anything with fine bright detail on dark: resize in **linear light**.
- Big enlargements you have decided to accept: a model-based upscaler, with the disclosure that goes with it.

Today: downscale one photograph of a fence or a striped fabric to 600 pixels twice, once with `-sample` and once with `-filter Lanczos -resize`. The moiré in the first one is the entire argument.

BEFORE YOU MOVE ON

A photograph of a night skyline is reduced from 4000 to 800 pixels wide in Photoshop with Bicubic Sharper. The small version looks distinctly darker and the lit windows have lost their punch, although the exposure was never touched. What is the mechanism?

A. The averaging happened on encoded numbers rather than on light, so mixed pixels land below the right luminance.
B. Bicubic Sharper adds an overshoot at edges, and the dark side of each overshoot has dragged the overall brightness down.
C. Reducing an image discards pixels, and the discarded ones happened to be the bright ones.
D. The image needed to be converted to sRGB before resizing so the numbers would be interpreted correctly.

69 · 9 min

The width to export, and why the browser will not fix it for you

THE ONE THING TO KEEP

Export width is the slot's CSS width times the device pixel ratio, capped at 2x for photographs; handing the browser a larger file costs the full bytes, is downscaled without gamma correction and throws away the sharpening you tuned for the output size.

Two kinds of pixel, and the number in between

A layout is written in **CSS pixels**, which are a unit of apparent size, not of hardware. A phone that reports a 390-pixel-wide viewport has around 1170 physical pixels across the glass. The ratio between them is the **device pixel**

ratio, and it is 1 on an old monitor, 2 on most laptops and mid-range phones, and 3 on many flagship phones.

That is the whole of the sizing arithmetic. An image displayed in a 600 CSS-pixel column needs 1200 real pixels to be crisp on a DPR 2 screen.

```
export width = CSS width of the slot x device pixel ratio
```

Work out the CSS width honestly. It is not the width of the page, it is the width of the box the image sits in at the breakpoint you care about — which is usually the phone, because that is where most of the traffic is and where the data costs money.

Stop at 2x

You could export at 3x for the DPR 3 phones. Almost nobody should.

A 3x file has 2.25 times the pixels of a 2x file, and therefore roughly 2.25 times the bytes. The visible gain over 2x on a screen held at normal reading distance is at the edge of what most people can detect, because at that pixel density the limit is your eye rather than the file. Some teams do serve 3x for logos and line art, where the hard edges do show it, and 2x for photographs. That is a defensible split.

The opposite mistake costs more. A 4000-pixel photograph dropped into a 600-pixel column is around 40 times the necessary data, and on a slow connection it is the difference between a page that loads and a page somebody leaves.

What the browser does with a mismatch, and why it is not free

Give a browser an image larger than the slot and it will scale it down. People take this as a reason not to worry. Three reasons to worry:

It downloads all of it first. The scaling is free. The bytes are not.

The downscale is not gamma-correct. Browsers resample in encoded space, so the darkening described in the resampling lesson applies to every oversized image you hand them, most visibly on fine bright detail.

There is no output sharpening. Your sharpening was tuned for the size you exported. Shrunk by an arbitrary factor in the browser, it is either invisible or crunchy. Resize to the delivered size yourself, then sharpen, and the result is noticeably crisper than the same photograph delivered oversized — at a fraction of the bytes. This is the single most common reason a site's images look slightly soft.

The mirror-image mistake is an image smaller than its slot, which the browser enlarges by interpolation with no detail to work from. A 400-pixel logo in an 800-pixel header is soft, and no CSS fixes it.

Numbers that are actually useful

These are starting points for a typical content site, at 2x, before you measure your own layout:

- Full-bleed hero on a phone: **800 to 1000 px** wide. On a wide desktop layout, 1600 to 2000.
- Article body image in a readable text column: **1200 to 1400 px**.
- Card or grid thumbnail: **600 to 800 px**.
- Avatar at 48 CSS px: **96 px**, and not one pixel more.
- Product image with a zoom feature: the zoom is the constraint, not the page. 2000 px square is the common floor.

Measure rather than guess. In any browser, right-click an image, *Inspect*, and the element panel shows both the natural size and the rendered size. If the natural width is more than about 2.5 times the rendered width, you are wasting somebody's data.

Aspect ratio, and the shift nobody notices until it ships

Export at the ratio the slot uses. If the layout crops a 3:2 photograph to 16:9 with `object-fit: cover`, the crop happens at the centre by default and it will cut heads off. Either crop deliberately in the editor and export at the target ratio, or set `object-position` so the crop favours the part that matters.

Always give the image element its `width` and `height` attributes, even when CSS resizes it. The browser uses them to reserve space before the file arrives, and without them the page jumps as each image loads — the thing that makes you tap the wrong link on a phone.

The free route

None of this needs a paid tool. GIMP's *Image > Scale Image*, Photopea's *Image > Image Size*, or one command:

```
magick in.jpg -colorspace RGB -filter Lanczos -resize 1200x -
colorspace sRGB \
  -unsharp 0x0.6+0.7+0.02 -quality 82 -strip out.jpg
```

Resize, sharpen at the delivered size, compress, strip metadata. That is the whole export in one line, and the order in it is the order from the sharpening lesson.

Today: open your own site or a page you made, inspect three images, and write down natural size against rendered size. The first number is usually three to six times the second, and each one of those is a file you can shrink to-day without anybody seeing a difference.

BEFORE YOU MOVE ON

A designer exports every article photograph at 3000 px wide and lets CSS fit them into a 700 CSS-pixel column. The page is slow, and the images also look slightly soft compared with a competitor's. Why the softness, given that the files contain far more detail than the slot needs?

A. Three thousand pixels exceeds what most screens can display, so the browser must discard detail and always renders oversized images at reduced quality.
B. The sharpening was tuned for 3000 px, and the browser's downscale neither preserves it nor replaces it at the delivered size.
C. JPEG compression artefacts become more visible when an image is scaled down in the browser rather than in an editor.
D. The images are missing width and height attributes, so the browser guesses the dimensions and rounds the scaling factor.

70 · 9 min

Sharpening adds no detail, and it goes last

THE ONE THING TO KEEP

Sharpening only exaggerates edges that already exist, so do it after resizing, at the output size, and after noise reduction rather than before.

What Unsharp Mask actually does

The name is backwards on purpose, and it tells you the mechanism. The software makes a blurred — unsharp — copy of your image, subtracts it from the original, and what remains is a map of the edges. It then adds that edge map back at increased strength.

The result is that along every edge, the light side is made lighter and the dark side darker. That local contrast increase is what your eye reads as sharpness.

No detail has been added. Nothing that was not in the file is now in the file. You have exaggerated the transitions that were already there.

Three controls, and they mean exactly what the mechanism implies:

- **Amount** — how much of the edge map is added back. 60 to 120 per cent for most photographs.
- **Radius** — how wide a blur was used, and therefore how thick the resulting bright and dark rims are. Small, 0.5 to 1 pixel, for fine detail. Large radius over about 3 is not sharpening any more, it is local contrast, and it makes skies look dramatic and skin look grubby.
- **Threshold** — how different two neighbouring pixels must be before the tool treats them as an edge. Set it to 0 and you sharpen the noise in a clear sky along with everything else. 2 to 4 protects flat areas.

Over-sharpen and you get **halos**: a pale outline around dark branches against a sky, and a dark outline around bright ones. Once you have seen one you will see them everywhere, including on television.

High Pass is the same idea with better control. Duplicate the layer, run *Filter > Other > High Pass* at 1 to 2 pixels, set the layer blend mode to Overlay or Soft Light, and adjust the layer opacity to taste. Because it is a layer, it takes a mask, so you can sharpen the eyes and not the skin. GIMP, Krita and Photopea all have both Unsharp Mask and High Pass.

Why it goes last

Every operation you do after sharpening changes the pixel scale that the sharpening was tuned for.

Sharpen a 4000-pixel-wide image, then resize it to 800 for a website, and the halos you created were sized for 4000 pixels. Shrinking the image does not shrink them proportionally in a clean way — the resampling stacks neighbouring halo pixels together, and the small version comes out crunchy with pale rims around every branch. Do it the other way round: resize to 800, *then* sharpen at 0.5 pixel radius, and the same photo looks clean.

The same photograph delivered at 800 pixels, two ways

Sharpened at 4,000, then resized

- The halos were sized for a 4,000-pixel frame
- Resampling stacks neighbouring halo pixels together
- Pale rims round every branch, and flat areas that crawl
- The noise was sharpened before anything reduced it

Resized to 800, then sharpened

- Radius 0.5 to 1 pixel, tuned to the delivered size
- Threshold 2 to 4, so a clear sky is left alone
- Chroma noise reduced generously, luminance sparingly
- High Pass on its own layer, so the skin can be masked out

Sharpening exaggerates edges that already exist, so everything done after it changes the scale it was tuned for. Resizing, noise reduction and compression all touch the pixels, which is why sharpening is last and at the size the picture will actually be seen.

Same argument for noise reduction, which smooths pixels, and for JPEG compression, which quantises them. Sharpening amplifies whatever is there, including compression artefacts and noise. It goes at the end, after everything else, at the size the picture will actually be seen.

Three sharpenings, not one

Professionals sharpen up to three times for different reasons, and it helps to know which one you are doing:

1. **Capture sharpening.** A small global correction for the softness introduced by the sensor's filter array and the demosaic process. Applied at full size, gently. Raw developers do this by default — darktable and RawTherapee both apply a modest amount unless you turn it off.
2. **Creative sharpening.** Local and masked. Eyes, jewellery, the edge of a product, the texture of fabric. Never the whole frame.
3. **Output sharpening.** At the final pixel dimensions, tuned to the medium. Print needs noticeably more than screen, because ink spreads slightly into paper — this is why an image that looks correctly sharpened on your monitor prints soft. Matte paper needs more than gloss.

Noise: two kinds, two different costs

Zoom into a photo shot at ISO 6400 and you see two things.

Chroma noise is the coloured speckle — random red, green and blue blotches in what should be a smooth grey. Removing it is nearly free, because real colour information in a photograph changes slowly across the frame. You can push chroma noise reduction hard and lose almost nothing.

Luminance noise is the grain-like variation in brightness. Removing it is expensive, because at the pixel level luminance noise and fine texture are the same thing: small local variations in brightness. The algorithm cannot tell the difference between grain in a sky and the weave in a shirt. Push the slider and skin turns to wax, grass turns to green paint, hair becomes a smooth mass. This is the plastic look, and it is far more damaging to a photograph than the noise was.

So: chroma reduction generously, luminance reduction sparingly, and mask it off the areas with real texture. Then sharpen. Denoise-then-sharpen is the fixed order; the reverse sharpens the noise and then smooths the detail.

The free route is strong here. darktable's profiled denoise uses measured noise profiles for specific camera and ISO combinations and outperforms most generic sliders. RawTherapee's is also good. Both cost nothing.

AI denoise, and the same warning as upscaling

Model-based denoisers — Lightroom's Denoise, DxO DeepPRIME, several open-source options — do not smooth. They reconstruct. Trained on pairs of noisy and clean images, they predict what the clean version should look like, and on high-ISO photographs the results are startling: two or three stops of apparent improvement.

The warning is the same one from the first lesson. What comes back is plausible, not recovered. Fine repeated texture — fabric weave, distant foliage, small text on a sign — can come back as a convincing pattern that is not the pattern that was there. For a portrait or a wedding, nobody cares and the result is better. For anything documentary or evidential, you have altered the record.

Today: take one photo, save two exports of it at 1000 pixels wide — one sharpened before the resize, one after. Look at them side by side at 100 per cent. You will not need to be told again.

BEFORE YOU MOVE ON

A landscape is sharpened at full 4000-pixel width, then resized to 800 pixels wide for a website. The small version looks crunchy, with pale outlines around the branches. What is the mechanism?

A. The halos created by sharpening were sized for the large image, and the down-sampling stacked them together, so sharpening should have been applied after the resize at the output size.
B. The resize used bicubic rather than Lanczos, and a better resampling filter would have avoided the outlines.
C. JPEG compression at the smaller size introduced the outlines, so exporting PNG would fix it.
D. Noise reduction was applied before sharpening; reversing that order would have kept the edges clean.

71 · 10 min

Why your image looks washed out on some-body else's phone

THE ONE THING TO KEEP

Deliver sRGB by converting rather than assigning, choose the format by what is in the picture, and treat file size as part of the job.

Pick the format by what is in the picture

Six formats cover everything you will do. The choice is not a matter of taste.

- **JPEG** — lossy, 8-bit, no transparency. It compresses by throwing away detail your eye is least likely to miss. Correct for photographs, and only for photographs. Every save re-compresses, so never edit and re-save the same JPEG repeatedly.

- **PNG** — lossless, supports transparency. Correct for screenshots, logos, flat graphics, anything with hard edges and large flat areas, and anything needing an alpha channel. Wrong for photographs: a PNG photo is routinely three to six times the size of a visually identical JPEG, and PNG photographs are one of the biggest causes of slow pages on slow connections.
- **WebP** — does both lossy and lossless, supports transparency, typically 25 to 35 per cent smaller than JPEG at matched quality. Supported by every current browser since around 2020. The sensible default for the web today.
- **AVIF** — smaller again, especially at low bitrates, supported across the major browsers by about 2024. Slower to encode, and it can smear very fine detail at aggressive settings, so check it on a texture-heavy image.
- **TIFF** — lossless, 16-bit capable, huge. For print delivery and archives.
- **PSD / XCF / KRA** — working files with layers. Not delivery formats. Do not send one to a printer expecting it to work.

The red-text problem

JPEG stores brightness at full resolution but usually stores colour at half resolution in each direction. That is **chroma subsampling**, written 4:2:0, and it is invisible in a photograph of a face or a landscape because colour genuinely does change slowly there.

It is very visible on saturated coloured text and hard colour edges. Red type on white saved as JPEG gets a soft, dirty fringe. That is not a quality-setting problem; it is structural to how the format stores colour.

So red text on white goes out as PNG or WebP, never JPEG. Screenshots of code, spreadsheets, anything with thin coloured lines: the same. This one rule will save you an argument with a client who thinks you delivered a blurry file.

Convert, do not assign

Now the question in the title.

A colour profile tells software what the numbers in your file mean. Red at 255 is a different physical red in sRGB than in Adobe RGB, because Adobe RGB describes a larger range of colours — a wider **gamut** — with the same 0-to-255 scale.

So here is the washed-out mechanism, exactly. You edit in Adobe RGB. You export with the Adobe RGB profile attached. The file goes to a marketplace, a chat app or an older Android gallery that ignores or strips the embedded profile. That software assumes sRGB, because everything assumes sRGB. Your numbers are now read against a smaller gamut, every saturated colour is pulled inwards, and the image looks flat and dull compared with what you saw.

The reverse also happens: an sRGB file interpreted as Display P3 comes out garish, with orange skin.

Two menu items sit next to each other and do opposite things:

- **Assign Profile** changes the label without changing a single number. The appearance shifts, because you have changed what the numbers mean.
- **Convert to Profile** recalculates every number so the appearance stays as close as possible in the new space.

For anything going onto the internet: **Convert to Profile > sRGB IEC61966-2.1**, then export with *Embed Colour Profile* ticked. In Photoshop's *Export As* or *Save for Web* there is a checkbox called "Convert to sRGB" — leave it on. GIMP has *Image > Colour Management > Convert to Colour Profile*. Photopea's colour management is limited, which in practice means it mostly works in sRGB and that is usually fine for web work and a liability for print.

Edit in a wide gamut if your monitor shows one and you know why. Deliver sRGB. Print is a different conversation — CMYK, paper profiles, soft proofing — and belongs in vector and print.

Sizes that matter on a slow connection

A large share of the people who will open your image are on a phone on mobile data, paying per megabyte. Treat file size as part of the design.

- **Resize to the size it will be displayed.** A 4000-pixel photo shown in a 600-pixel column is 40 times more data than necessary. If the site serves high-density screens, export at twice the display width and stop there.
- **Quality 100 is a waste.** JPEG quality 75 to 82 is typically indistinguishable from 100 at a third of the size. Test it: export both, flip between them at 100 per cent zoom, and see whether you can tell.
- **Aim for 100 to 300 KB** for a full-width photograph on a page. A 3 MB hero image is a decision to lose readers.
- **Check the actual number.** Squoosh (squoosh.app) shows original and compressed side by side with a live file size, in a browser, free. If it has gone by the time you read this, ImageMagick and `cwebp` do the same from a command line, and GIMP's export dialogue shows the size as you move the quality slider.

Metadata, including the part about your address

A photo from a phone carries EXIF: camera, settings, date, and often GPS coordinates. Posting a picture of something you are selling, taken in your home, with the coordinates attached, tells every stranger where you live.

Most export dialogues have a metadata dropdown. Set it to *None* or *Copyright only* for anything public; `exiftool -all= photo.jpg` strips the lot. Some platforms remove EXIF on upload, several do not, and a file sent as a document rather than an image keeps everything.

Today: check your last exported image — pixel width, file size, colour profile. If it is 4000 pixels and 4 MB for a web page, you have found a five-minute fix worth more than an hour of retouching.

BEFORE YOU MOVE ON

A photographer edits in Adobe RGB and exports with that profile embedded. On a marketplace listing the colours look dull and flat compared with the editor. What is the correct account of it, and the fix?

A. The marketplace re-compressed the JPEG, and the loss of quality reads as flatness; exporting at a higher quality setting will fix it.
B. The editing monitor is wide gamut and phone screens are not, so nothing can be done about it.
C. The image needs its saturation raised before export to compensate for what the platform does to it.
D. The platform ignored the embedded profile and read the same numbers as sRGB, which maps saturated colours inwards; converting to sRGB before export re-maps the numbers so the appearance survives.

72 · 9 min

Stop guessing the quality slider

THE ONE THING TO KEEP

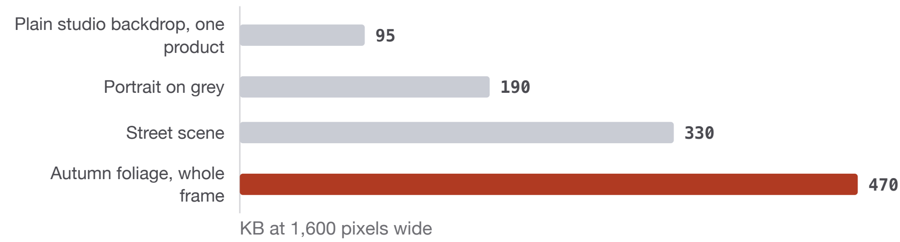
A fixed quality setting gives different visual results on different pictures because it scales a quantisation table rather than targeting an outcome, so pick settings by flipping between versions at 100 per cent, automate that judgement with a perceptual metric, and argue in byte budgets rather than slider numbers.

One number cannot serve forty images

The usual method is to pick a quality setting once — 80, say — and apply it to everything. It is better than quality 100, and it is still wrong, for a reason that follows directly from how the encoder works.

The quality setting scales a quantisation table. It does not target a file size or a visual result. Feed it a photograph of a plain studio background and quality 80 produces a small file that is visually perfect. Feed it a crowd scene in autumn foliage and the same setting produces a large file that still shows artefacts in the leaves, because there is far more fine detail competing for coefficients.

The same JPEG quality 82, on four different pictures



The setting scales a quantisation table. It does not target a file size and it does not target a visual result, so one number across a set gives you files that vary five-fold in weight and still vary in how good they look. jpeg-recompress finds the lowest quality per file that stays under a perceptual threshold, and typically saves a quarter to two fifths with no image worse.

So a fixed quality across a set gives you files that vary wildly in size **and** vary in how good they look. That is the worst of both.

The honest visual test, done properly

Before any tooling, learn to do this by eye, because it is the ground truth everything else approximates.

1. Export the same image at three settings — say 90, 78 and 65.
2. Open all three at **100 per cent zoom**, not fit-to-window. A scaled view hides artefacts and invents others.
3. Flip between them in the same screen position. Do not view them side by side; your eye is far better at spotting change than at comparing two things at once.
4. Look where JPEG fails: flat gradients such as sky and studio backdrops, shadow areas, hard edges against flat colour, and any saturated red.
5. Take the lowest setting where flipping shows you nothing.

Most photographs land between 72 and 85. Images with large smooth gradients need more; textured images tolerate less than people expect because the artefacts hide in the texture.

Metrics, and what each one is actually measuring

File size difference is not quality. Nor is PSNR, the signal-to-noise figure you will see quoted, which counts every altered pixel equally and therefore rewards blurring.

SSIM and its distance form **DSSIM** compare local structure — means, variances and correlation over small windows — and correlate much better with what people notice. `dssim a.png b.png` returns a number where 0 is identical; under about 0.001 is usually invisible on a photograph.

Butteraugli, and the **ssimulacra2** score built on similar ideas, model the human visual system more directly, including how much distortion a busy region hides. They are the current best free proxies. Butteraugli returns a distance where lower is better and roughly 1.0 is the threshold of a just-noticeable difference.

The point of a metric is not to replace your eyes. It is to let you run the decision over 400 files instead of 4.

Per-image quality targeting

This is the technique the fixed-slider approach is missing, and free tools do it.

`cjpeg` with a quality search, or more simply `jpeg-recompress` from the jpeg-archive package, encodes at several settings, measures each against the original with a perceptual metric, and keeps the lowest setting that stays under your chosen threshold:

```
jpeg-recompress --quality medium --method ssim in.jpg out.jpg
```

Every file comes out at the quality it needs. Across a real catalogue this typically saves 25 to 40 per cent against a fixed quality 82, with no image worse.

`avifenc` and **`cwebp`** both take a quality argument and `cwebp -size` will target a byte count directly. Squoosh, in a browser, shows the live file size and a slider and lets you do the flip test against the original in the same window, free, with nothing uploaded anywhere.

Budgets, which is how you actually decide

Quality settings are easier to argue about than to justify. Byte budgets are not.

- A full-width photograph on a page: **100 to 300 KB**.
- A grid thumbnail: **15 to 40 KB**.
- A whole article page, images included: aim under **1 MB**, and know that you are choosing when you go over.
- An email: under **100 KB** per image, because many clients do not lazy-load and several proxy and re-compress anything larger.

A budget converts an aesthetic argument into arithmetic. If the hero must be 1600 pixels wide and must fit 250 KB, the quality setting is whatever achieves that, and if nothing does, the picture is the wrong picture for the slot.

Two failure modes worth naming

Compressing an already-compressed file. Re-encoding a JPEG you were sent at quality 60 to quality 85 does not improve it; it stores the existing artefacts faithfully in a bigger file. Compress once, from the highest-quality source you have.

Optimising the wrong thing. Shaving 30 KB off a hero that is 2400 pixels wide for an 800-pixel slot is tidying a room that needs a wall moved. Get the dimensions right first; compression is the second-order saving.

Today: take five images from one project, run the flip test on each, and write down the lowest quality that passed. The spread between them is the reason a single number was never going to work.

BEFORE YOU MOVE ON

A shop exports 400 product photographs at a fixed JPEG quality of 82. The plain-background shots come out at 90 KB and look perfect; the shots on textured wooden tables come out at 600 KB and still show mush in the grain. What does this tell you about the setting?

A. The textured images were photographed at a higher ISO, and noise is inflating both the file size and the artefacts.
B. Quality 82 is too low overall, and raising it to 92 would fix the textured shots at an acceptable cost.
C. The setting scales the quantisation coarseness, not the outcome, so detailed images cost more and still lose more.
D. The wooden-table shots should have been exported as PNG, since JPEG cannot represent fine repeated texture.

73 · 10 min

The formats that replaced JPEG, and the one that might

THE ONE THING TO KEEP

WebP and AVIF are smaller than JPEG because they use variable block sizes and intra prediction borrowed from video codecs, AVIF's characteristic failure is smoothing texture rather than blocking it, JPEG XL can losslessly shrink existing JPEGs but its browser support is genuinely unresolved, and quality numbers never transfer between encoders.

Why the newer formats are smaller

JPEG splits an image into fixed 8x8 blocks and transforms each one independently. That was a sensible design in 1992 and it leaves a great deal on the table.

WebP and **AVIF** are both built from video codecs — VP8 and AV1 respectively — and they bring the tricks video encoders use inside a single frame. The important ones:

- **Variable block sizes.** A flat sky is described with a few large blocks; a face gets small ones. JPEG spends the same 8x8 grid on both.
- **Intra prediction.** Before transforming a block, the encoder predicts it from the pixels already decoded above and to the left, in one of several directions, and only stores the difference. On a gradient or a smooth surface the difference is nearly nothing.
- **Better entropy coding.** Arithmetic coding rather than JPEG's Huffman tables, which is a few more per cent for free.

The practical result at matched visual quality: WebP is typically 25 to 34 per cent smaller than JPEG, AVIF another 20 to 30 per cent smaller than WebP, with the gap widening at low bitrates and narrowing at high ones.

What each one is good and bad at

WebP. Lossy and lossless in one format, alpha in both, animation. Supported by every current browser since 2020 and by most tooling. Its lossy mode inherits VP8's chroma handling and is always 4:2:0, so saturated red edges suffer the same fringing as JPEG. Its lossless mode is genuinely good on graphics and often beats PNG by 20 to 30 per cent.

AVIF. The best compression of the three at web sizes, real 10- and 12-bit support, alpha, wide gamut and HDR, animation. Two costs. Encoding is slow — seconds per image at high effort settings rather than milliseconds — and at aggressive settings it **smooths** rather than blocks. That is a more insidious failure than JPEG's: fine grain, distant foliage and fabric weave quietly become flat, and the image looks clean rather than damaged. Always check a texture-heavy region at 100 per cent before committing a whole catalogue to AVIF.

JPEG XL. Technically the most interesting of the three. It can **losslessly transcode an existing JPEG** to roughly 80 per cent of its size, and turn it back byte-for-byte — so an archive of a million JPEGs can shrink by a fifth with nothing lost and nothing re-encoded. It supports progressive decoding properly, high bit depths and very high quality at low cost.

And its browser support is unsettled. Safari ships it. Chrome implemented it, removed it in 2023, and the argument about whether to restore it has run ever since; Firefox has kept it behind a flag. Any confident statement about where this lands is a guess. Check the current state before you rely on it, and do not build a delivery pipeline whose only output is JPEG XL today.

Encoder settings that matter

```
cwebp -q 78 -m 6 in.png -o out.webp
avifenc --min 0 --max 63 -a end-usage=q -a cq-level=28 -s 4 in-
.png out.avif
cjxl in.jpg out.jxl          # lossless transcode of an exist-
ing JPEG
```

Two things to understand in there.

Effort or speed settings are not quality settings. `-m 6` in cwebp and `-s 4` in avifenc tell the encoder to search harder for a better encoding. Higher effort means a smaller file at the *same* quality, paid for in seconds. For a one-off hero image, use the slowest setting. For 5000 thumbnails, do not.

Quality numbers do not transfer between formats. WebP 78 is not AVIF 78 is not JPEG 78. Each encoder has its own scale — AVIF's `cq-level` even runs the other way, lower being better. The only way across is a perceptual measurement or your own eyes, which is the previous lesson.

Delivering more than one format

You do not have to choose one. The `<picture>` element lets the browser take the first source it understands:

```
<picture>
  <source srcset="photo.avif" type="image/avif">
  <source srcset="photo.webp" type="image/webp">
  <img src="photo.jpg" width="1200" height="800" alt="...">
</picture>
```

The JPEG stays as the floor. Anything that does not understand the newer types — an old email client, a scraper, somebody's image-saving script — still gets a file that works.

Where the old formats are still right

- **Anything a stranger will download and open in unknown software.** A JPEG opens everywhere, including on a fifteen-year-old machine in an office you will never see.
- **Print delivery.** A lab or a press wants TIFF or JPEG. Sending AVIF to a print shop will get you a phone call.
- **Editable archives.** None of these are working formats. Keep the layered master.
- **Email.** Support across mail clients is far patchier than across browsers, and some gateways re-encode anyway.

Today: take one photograph with visible grain or foliage, export it as JPEG at your usual setting, then as WebP and AVIF at file sizes 30 per cent smaller, and compare the textured area at 100 per cent. You are looking for whether the AVIF has kept the grain or quietly painted over it.

BEFORE YOU MOVE ON

An editor converts a catalogue of tweed jacket photographs to AVIF at an aggressive setting and reports that they look cleaner than the JPEGs. A buyer later complains the fabric looks like felt. What happened?

A. AVIF's alpha channel was discarded during conversion, flattening the shadow detail that gave the fabric its depth.
B. The AVIF encoder ran at a low effort setting, and effort settings trade file size for detail retention.
C. At low bitrates AVIF smooths fine texture rather than blocking it, so lost weave reads as clean.
D. AVIF uses 4:2:0 chroma subsampling, which removes the colour variation that makes tweed read as woven.

74 · 9 min

GIF is a 1987 format and you are probably still exporting one

THE ONE THING TO KEEP

GIF costs ten to twenty times what the same motion costs as video because it stores 256 dithered colours per frame with no motion compensation, so deliver MP4 or WebM with muted autoplay, or animated WebP where an image tag is required, and keep GIF only for platforms that accept nothing else.

Why the animated GIF is so large

GIF was designed in 1987 for dial-up bulletin boards, and two of its design decisions are why a three-second clip becomes a four-megabyte file.

It has 256 colours per frame. Every frame is an index into a small palette. A photograph has far more than 256 distinct colours, so the encoder must either post it to the nearest palette entry — producing visible banding — or

dither, scattering pixels of two palette colours to fake a third. Dithering looks acceptable and it is catastrophic for file size, because it replaces smooth areas that would have compressed to nothing with high-frequency noise that compresses to almost nothing saved.

It has no motion compensation. Modern video codecs describe a frame as *this block moved eleven pixels left*. GIF can skip unchanged regions between frames and that is all. Pan the camera and every pixel changes, so every frame is stored nearly in full.

The result, for a typical 3-second 800-pixel clip: GIF around 4 MB, the same clip as H.264 MP4 around 250 KB. Sixteen times the data for a worse picture.

The four things you can deliver instead

- **MP4 / H.264 video.** Smallest by a wide margin, full colour, hardware-decoded so it costs almost no battery. Use it for anything longer than about a second or with real motion.
- **WebM / VP9 or AV1.** Smaller again, supports real alpha transparency, which MP4 effectively does not. This is how a transparent animated logo gets onto a page.
- **Animated WebP.** Full colour, alpha, and roughly 20 to 40 per cent the size of the equivalent GIF. Supported by current browsers. Convenient because it is still an image, so it drops into an `<img>` tag with no player.
- **APNG.** Full colour, proper alpha, lossless. Larger than animated WebP but the right answer for pixel-exact UI animation and small loops where lossy compression would show.

Making a video behave like a GIF

The reason people reach for GIF is that it plays by itself. A video can do the same:

```
<video autoplay muted loop playsinline width="800"
height="450">
  <source src="clip.webm" type="video/webm">
  <source src="clip.mp4" type="video/mp4">
</video>
```

All four attributes are load-bearing. `muted` is required or mobile browsers refuse to autoplay at all. `playsinline` stops iOS taking it fullscreen. `loop` replaces the GIF's loop flag. `autoplay` without `muted` silently does nothing, which is the commonest reason a developer says the video does not work.

Conversion, free, one line:

```
ffmpeg -i in.gif -movflags faststart -pix_fmt yuv420p \
  -vf "scale=trunc(iw/2)*2:trunc(ih/2)*2" out.mp4
```

The `scale` filter is there because H.264 requires even dimensions and a 801-pixel GIF will otherwise fail with an unhelpful error. `faststart` moves the index to the front of the file so playback can begin before the whole thing arrives.

When GIF is still the answer

Honest cases remain, and they are all about where the file is going rather than what is in it.

Messaging apps, older email clients, some chat platforms and a number of content management systems accept an image and will not accept a video. A GIF plays there; an MP4 becomes an attachment nobody opens. Platform behaviour, not technical merit, is the reason.

If you must ship one, cut the cost deliberately: reduce to 480 pixels wide, drop to 12 or 15 frames per second, keep it under two seconds, and use `gifsicle -O3 --lossy=80` or `gifski`, which builds a far better palette per frame than most exporters. A disciplined GIF is 600 KB rather than 4 MB.

Cinemagraphs, which are a raster editing job

The most useful moving image an image editor makes is the one where almost nothing moves: steam rising from a cup in an otherwise frozen photograph, a single flag moving, the rest still.

The construction is exactly the mask work from module three, applied to time. Take a locked-off clip on a tripod, pick one frame as the still base, put the video layer above it, and mask the video so only the region that should move is visible. Because the static 95 per cent of the frame is genuinely identical between frames, every codec compresses it to nearly nothing — a cinemagraph is often smaller than a still-plus-video would suggest, and far more striking than a full-motion loop.

Free path: shoot on a phone on a tripod or propped against a book, then GIMP with GAP, Krita's animation workspace, or Blender's video sequencer, all of which will hold a masked layer over a clip. DaVinci Resolve's free edition does it comfortably.

The accessibility rule that is also a legal one in places

Anything that flashes more than three times per second can trigger a seizure. This is not a style preference; it is in the Web Content Accessibility Guidelines and it is referenced by accessibility law in several countries.

Give any autoplaying loop longer than about five seconds a way to stop it, and never build strobing or rapidly inverting animation. If a client asks for one, say why not.

Today: take any GIF you have, run the ffmpeg command above, and compare the two file sizes. The ratio is usually between ten and twenty to one, and it is the clearest argument you will ever have with somebody who wants a GIF.

BEFORE YOU MOVE ON

A team replaces a 4 MB product GIF with an MP4 in a video element carrying autoplay, loop and playsinline. On desktop it plays; on phones nothing happens at all. What is missing?

A. The faststart flag, without which a mobile browser waits for the entire file before it can begin decoding the first frame.
B. A WebM source, because mobile browsers will not autoplay H.264 without a royalty-free fallback available.
C. The muted attribute, which mobile browsers require before they will autoplay anything.
D. Explicit width and height attributes, since a video with no reserved space is skipped during layout on small screens.

75 · 8 min

The fields that say who made it, and which platforms keep them

THE ONE THING TO KEEP

EXIF records the camera, IPTC records the picture's meaning and XMP is the container modern tools write, so fill Creator, Copyright, Credit and Description as an import preset — and treat embedded metadata as a record that most social re-encoding destroys rather than as any kind of protection.

Three standards, one file

Open any photograph in `exiftool` and three different metadata systems come out at once.

EXIF is the camera's record: make, model, lens, shutter, aperture, ISO, date, orientation, and often GPS. Written by the device, rarely edited by hand.

IPTC is the press standard, defined for news agencies in the 1990s and still the one picture desks, stock libraries and image databases read. Its fields are about the *picture's* meaning rather than the camera's settings: Creator, Credit Line, Copyright Notice, Headline, Description, Keywords, City, Country.

XMP is Adobe's container, an XML block that can carry IPTC fields, editing history, ratings and labels, and anything else a tool wants to embed. It is the one modern software actually writes, and it can live alongside the file as a sidecar `.xmp` for formats that cannot embed it, notably raw files.

The practical consequence: when people say "the copyright is in the file", they usually mean an IPTC field carried inside an XMP block. Fill it in one place and most software shows it.

The four fields worth filling every time

You do not need the whole schema. Four fields carry nearly all of the value.

- **Creator** — your name, or the photographer's. This is what a picture desk searches on.
- **Copyright Notice** — `© 2026 Your Name`. Its presence is what makes removal an identifiable act rather than an oversight.
- **Credit Line** — how you want to be named in print, which is not always the same as your legal name.
- **Description** — one sentence saying what the picture shows. This one earns its keep twice: it is what a database search matches, and it is the starting text for the alt attribute when the image goes on a page.

Set them once as a preset. Lightroom, darktable and digiKam all apply a metadata template on import, so every frame from every shoot carries your name without a further thought. On the command line:

```
exiftool -Creator="A. Photographer" -CopyrightNotice="© 2026 A.
Photographer" \
  -Credit="A. Photographer" -overwrite_original *.jpg
```

What the platforms actually do

This is the part that decides whether any of it matters, and it is worth testing rather than assuming, because behaviour changes without announcement.

Broadly: most large social platforms re-encode uploaded images, and re-encoding rebuilds the file from decoded pixels. Whatever metadata survives is whatever the platform chose to copy across, which is usually little or none. Some keep copyright and creator fields and drop everything else. Messaging apps typically strip the lot along with a good deal of resolution.

Sites whose business is images behave differently — stock libraries, photo-sharing services aimed at photographers and most content management systems preserve IPTC because their users would revolt otherwise.

The test takes four minutes and settles the question for the platform you care about: fill the fields, upload, download the file the platform serves back, and run `exiftool` on it. Do this again in a year.

The conclusion to draw is not that metadata is pointless. It is that **embedded metadata is a record, not a protection.** It survives the file being copied, emailed, archived and re-opened in another program. It does not survive a screenshot, and it does not survive most social uploads. Cryptographically signed provenance, which is the C2PA work in module seven, is an attempt to fix exactly this gap, and it breaks at the same re-encoding step unless the platform participates.

The half that must not survive

The export lesson made the privacy point about GPS. Two more cases are worth knowing.

Orientation. EXIF has a rotation flag, and some software applies it while some ignores it. A photograph that is upright in your editor and sideways on a client's machine is almost always this. Bake the rotation into the pixels before delivery — `exiftool -Orientation=1 -n` after a real rotation, or `magick -auto-orient` — rather than relying on every downstream reader to honour the flag.

Editing history. XMP can record the software, the edit steps, and sometimes earlier file paths, which have a habit of containing a client's name or your own directory structure. For an external deliverable, strip everything and write back only the four fields you want:

```
exiftool -all= -tagsfromfile @ -Creator -CopyrightNotice -
Credit -Description \
  -ColorSpaceTags -overwrite_original out.jpg
```

`-ColorSpaceTags` is the important qualifier. A blanket `-all=` removes the ICC profile too, and a file with no profile is read as sRGB by everything — which is fine if you converted to sRGB, and which turns an Adobe RGB file washed out in exactly the way the export lesson described.

Today: run `exiftool` on the last photograph you delivered to anybody. Note what it says about your name, your location and your software. Then decide which of those you intended to send.

BEFORE YOU MOVE ON

A photographer strips metadata from a set of Adobe RGB exports with `exiftool -all=` before sending them to a client. The client reports the images look flat and desaturated. What did the strip do?

A. It removed the IPTC description fields, and the client's viewer falls back to a generic rendering when a file carries no description.

B. It removed the ICC profile as well, so the wide-gamut numbers are now read as sRGB and every saturated colour is pulled inwards.

C. It removed the EXIF orientation flag, and the resulting re-render discarded colour information.

D. Stripping metadata forces a re-encode, and the re-encode was performed at a lower quality setting.

76 · 9 min

One picture, six files, and letting the browser choose

THE ONE THING TO KEEP

Ship several widths and formats and let the browser choose with srcset, sizes and picture — sizes being the attribute that actually prevents oversized downloads — then lazy-load everything except the hero, which needs high fetch priority because it is what the load-time metric is measuring.

The problem a single file cannot solve

The previous lessons give you a width for a destination. A web page has no single destination: the same image sits in a 360-pixel column on a phone, a 700-pixel column on a laptop and a 1200-pixel column on a desktop, across device pixel ratios of 1, 2 and 3.

Export for the largest and you punish every phone. Export for the phone and the desktop is soft. The answer is to ship several files and describe them, letting the browser pick before it downloads anything.

srcset and sizes, and which does what

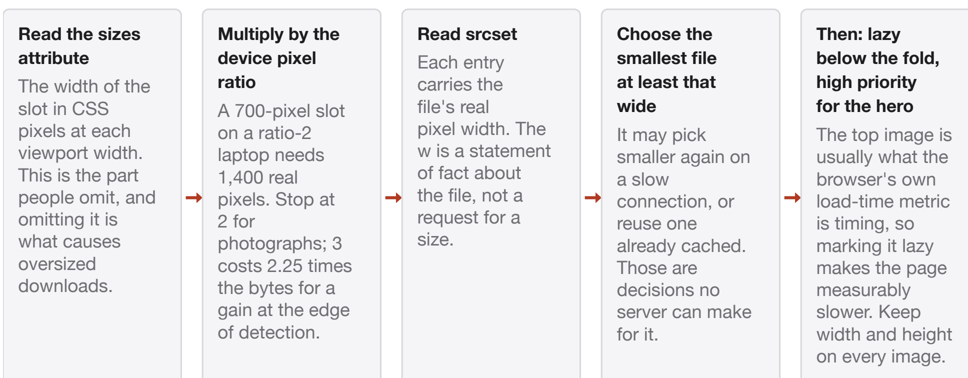
```
<img src="photo-800.jpg"
     srcset="photo-400.jpg 400w, photo-800.jpg 800w, photo-1600.jpg 1600w"
     sizes="(max-width: 700px) 100vw, 700px"
     width="1600" height="1067"
     alt="Brass taps on a workbench">
```

`srcset` lists the files and the real pixel width of each. The `w` is a fact about the file, not a request.

`sizes` tells the browser how wide the slot will be, in CSS pixels, at each viewport width. This is the part people leave out, and leaving it out means the browser assumes the image fills the viewport and downloads something much larger than it needs.

The browser then does the arithmetic you would have done: slot width times device pixel ratio, choose the smallest file at least that big. It may also choose a smaller one on a slow connection or when the file is already cached, which is a decision you cannot make from the server.

How the browser picks, before it downloads anything



Leave sizes out and the browser assumes the image fills the viewport, so it fetches something far larger than the slot and nothing visibly breaks. The attribute is written in CSS units but lives in the HTML, which is why a layout change quietly desyncs it and why most site builders now generate both from one definition.

`src` stays as the fallback for anything that does not understand `srcset`.

The one genuine annoyance: `sizes` is written in CSS units but lives in the HTML, so it has to be kept in step with a stylesheet it cannot see. When a layout changes and nobody updates `sizes`, the wrong file is served and nothing visibly breaks. Most static site generators and frameworks now write both from one definition, which is the real fix.

Combining formats and sizes

`<picture>` handles format choice, `srcset` handles size, and they compose:

```
<picture>
  <source type="image/avif" srcset="p-400.avif 400w, p-800.avif
800w, p-1600.avif 1600w" sizes="...">
  <source type="image/webp" srcset="p-400.webp 400w, p-800.webp
800w, p-1600.webp 1600w" sizes="...">
  <img src="p-800.jpg" srcset="p-400.jpg 400w, p-800.jpg 800w,
p-1600.jpg 1600w" sizes="..."
       width="1600" height="1067" alt="...">
</picture>
```

That is nine files for one photograph, which is why nobody writes this by hand. Generate it: a shell loop over `magick`, a `sharp` script, or whatever your site builder provides. The next module's derivative-set lesson is the production version of this.

`<picture>` with `media` attributes also gives you **art direction** — a genuinely different crop for narrow screens, not merely a smaller one. A wide landscape with the subject at the left becomes an unreadable strip on a phone; a square crop centred on the subject is a different photograph and the correct one.

Doing it without generating anything

An image CDN takes one master and serves derivatives from URL parameters:

```
https://cdn.example.com/photo.jpg?w=800&fm=avif&q=76
```

Cloudflare Images, imgix, Cloudinary and the image pipelines built into most hosting platforms all work this way, as does the free and self-hostable **imgproxy** and **Thumbor**. The saving is real and the cost is a dependency: every image on your site now loads through somebody else's service, on their pricing and their uptime. For a small site, pre-generating files at build time and serving them as static files is simpler, cheaper and faster.

Lazy loading, and the one image that must not be lazy

`loading="lazy"` tells the browser not to fetch an image until it is near the viewport. On a long page it is the single biggest saving available.

Apply it to the image at the top of the page and you make the page measurably slower. That image is usually the **Largest Contentful Paint** element — the thing whose arrival the browser's own performance metric is timing — and marking it lazy pushes its request behind others and delays the moment the page looks loaded.

The rule: everything below the fold gets `loading="lazy"`; the hero gets `fetchpriority="high"` and no lazy attribute. You can see the effect in any browser's developer tools under Lighthouse or Performance, free, in about thirty seconds.

And keep `width` and `height` on every image. A lazy-loaded image with no reserved space makes the page jump as the reader scrolls, which is how somebody taps the wrong thing.

Today: open a page you built, throttle the network to Slow 4G in developer tools, and reload. Watch which image arrives first. If it is a thumbnail from the footer, you have found the ordering problem this lesson is about.

BEFORE YOU MOVE ON

A developer adds a three-width \`srcset\` to every article image but omits \`sizes\`. Phones still download the 1600-pixel file although the images sit in a 340-pixel column. Why?

A. Without \`sizes\` the browser assumes the image spans the viewport, so it asks for far more than the slot needs.
B. The \`w\` descriptors describe intent rather than file width, so the browser cannot rank the candidates and takes the last one listed.
C. \`srcset\` only takes effect inside a \`picture\` element, and a bare \`img\` ignores it entirely.
D. Phones report a device pixel ratio of 3, so the browser multiplies the column width and legitimately needs the largest file available.

77 · 9 min

Your file is re-encoded on upload, and you can plan for it

THE ONE THING TO KEEP

Platforms resize, re-compress and strip the profile from every upload, so convert to sRGB, match the width they actually serve, sharpen slightly less than for final delivery, and give their encoder a high-quality input rather than one you already compressed.

You are not the last encoder

You spend an hour on an edit, export a careful 2400-pixel JPEG at quality 88, upload it, and it arrives soft, slightly grey and about 180 KB. Nothing went wrong in your export. The platform re-encoded it.

Essentially every consumer platform — social networks, marketplaces, messaging apps, forums, most content management systems — resizes and re-compresses on upload. They do it for bandwidth, for storage, and to normalise the wildly varying files their users send. Your file is an input to their pipeline, not the thing that gets served.

Once you accept that, the job changes: give their encoder the best possible input, and lose the least in their pass.

The three things their pipeline does

It resizes to a fixed width. Each platform has its own set of widths, and if your file does not match one, it is resampled to the nearest — usually without gamma correction and always without output sharpening tuned to that size. This is the largest single cause of the softness.

It re-compresses. Your quality-88 file is decoded and re-encoded, often more aggressively and often at 4:2:0 chroma subsampling regardless of what you chose. Everything the JPEG lesson says about generation loss applies, and you get one whole generation whether or not you edited anything.

It strips metadata. Usually including the ICC profile. A file that was not converted to sRGB before upload is then read as sRGB, and comes out flat in exactly the way the export lesson describes. This is why wide-gamut exports look dull on social and fine in your editor.

Some platforms also re-encode to their own format. Serving WebP or AVIF to browsers that accept them is now standard, so the file a reader receives may not even be the format you uploaded.

Matching their width is the whole trick

If you upload at a width the platform already uses, its resize step becomes a no-op or close to it, and your sharpening survives.

The widths move, so treat any list as something to verify rather than memorise. The method for verifying is fixed and takes five minutes: upload a test image, view the post in a desktop browser, right-click the displayed image and open it in a new tab, and read the dimensions and file size of what you were actually served. Do that once per platform you work with, write it down, and redo it when your results change.

What you will generally find: feed images on the large social networks are served somewhere between 1080 and 1440 pixels on the long edge; marketplaces are larger because of zoom, commonly 1600 to 3000; messaging apps are far smaller and far more aggressive than anything else, often under 1600 pixels and heavily compressed.

Giving their encoder a good input

Six things, in order of how much they help:

1. **Convert to sRGB and embed the profile.** Costs nothing, and prevents the most visible failure.
2. **Upload at the platform's own width** where you can determine it.

3. **Sharpen a little less than you would for final delivery.** Their re-encode will exaggerate halos, and a file sharpened for print arrives crunchy.
4. **Give it a high-quality JPEG or a PNG**, not a file you already squeezed to 90 KB. Compression artefacts in your input become the input to their compression, and the two stack. This is counter-intuitive — you are up-loading a larger file to get a smaller, better one back.
5. **Avoid fine dithered gradients and heavy grain**, which are what their encoder spends its budget on and loses first.
6. **Send as a document, not as a photo**, where the platform offers the distinction and the recipient needs the real file. Most messaging apps have this option, and it bypasses the whole pipeline.

The cases where none of this is enough

Some things cannot be pre-empted, and it is better to know than to keep fighting.

Repeated re-uploads. A picture downloaded from one platform and posted to another has been re-encoded twice, and the third generation is visibly de-graded. Always upload from your own master.

Screenshots of your work. Every screenshot is a fresh capture at some-body else's screen resolution with your metadata gone and no way to object.

Platform colour handling on phone apps. Some apps do not colour-manage at all on some devices, and a P3 display will show a saturated version of an sRGB file that you cannot correct for without wrecking it everywhere else. Deliver correct sRGB and let it be their bug.

Automatic enhancement. A few platforms and several phone gallery apps apply their own contrast or saturation adjustment on upload or display. There is no setting you can send that turns it off.

Today: upload one image you edited carefully, download the version the plat-form serves back, and open both at 100 per cent. The difference is the pipeline you are working through, and knowing its size changes how much effort the last five per cent of your edit deserves.

BEFORE YOU MOVE ON

An editor compresses images hard before uploading to a social platform, reasoning that a smaller input means less for the platform to degrade. The posted images look worse than a colleague's, who uploads large high-quality files. Why?

A. Large uploads are given priority in the platform's encoding queue and receive a higher quality setting.
B. Their encoder re-compresses what it receives, so your artefacts are encoded again and the two losses stack.
C. Small files are upscaled to the platform's fixed width, and upscaling is what introduces the visible damage.
D. Heavier compression removes the ICC profile, so the pre-compressed files are interpreted in the wrong colour space.

78 · 12 min

One bottle, one window, twelve minutes

THE ONE THING TO KEEP

The order is fixed — shoot, develop, straighten, mask, retouch, tone, then resize, sharpen, convert to sRGB and export — and the layered file is saved before any of it.

The job, and the light it needs

Someone sells 500 ml jars of honey and needs a listing photograph. No studio, no lights, no Photoshop subscription. A phone, a window, and a laptop that struggles with anything heavy.

The whole job, in order, with the actual numbers. Every tool named has a free path. Ten minutes on the shoot saves an hour of editing.

- A table beside a window, not in direct sun. An overcast day is the best free light there is.
- A white bedsheet or A2 card behind and under the jar, curving up the wall so there is no corner line.
- A second sheet of white card on the shadow side, about 40 cm away, bouncing light back. That is the difference between a product shot and a snapshot.
- **Wipe the lens.** Then wipe the jar.
- Shoot from the height of the product, not from above. Standing over it makes it look small.
- Use the phone's Pro mode and shoot **DNG** if it offers one. Most mid-range Android phones do; iPhone Pro models shoot ProRAW. Raw gives 12 or 14 bits per channel instead of 8, which is what lets you fix white balance later without banding.
- Tap to focus on the label, then drag the exposure down a third of a stop. Blown highlights on glass are unrecoverable; slightly dark is fixable.
- Take twenty frames. Storage is free.

From raw file to working file

Open the DNG in **RawTherapee** or **darktable** — both free, both fine on a modest laptop — or Lightroom mobile's free tier on the phone itself.

- **White balance**: take the eyedropper and click the white sheet — not the brightest highlight, which is probably clipped and useless as a reference. The window's colour cast disappears in one click.
- **Exposure**: nudge until the histogram's right edge stops just short of the wall.
- **Lens correction**: on. Phone wide lenses barrel noticeably.
- Export a **16-bit TIFF**, not a JPEG. You are about to stretch this file.

Then in Photopea, GIMP or Photoshop: straighten on the table's horizontal — a jar leaning two degrees reads as amateur before anyone can say why — and crop square at 2000 by 2000, above the 1600-pixel minimum several market-places set.

Save the layered file now, as `honey-500ml_front_v1.psd`, before any edit. That is the habit from lesson two, and the one people skip.

Cut it out, then clean it

Hard manufactured edges are the pen tool's job.

- Trace the silhouette with the pen tool. Ten to fifteen anchor points, not fifty. Sit it half a pixel *inside* the edge so no rim of grey background survives.
- Convert it to a selection, feather 0.5 px, and click **Add Layer Mask**. A mask, not Delete.
- Put a solid white layer underneath. Several marketplaces require the main image on pure white, 255/255/255; on your own site use about 250 grey, so the jar's highlights do not dissolve into it.
- **Keep the contact shadow.** Paint the dark pool where the jar meets the surface back in, on the mask, with a soft black brush at 20 per cent. Without it the jar floats and the picture looks fake to people who cannot say why. If the real shadow is unusable, paint a soft ellipse on a new layer set to Multiply at 20 to 25 per cent. That is fabricating light, not the product.

Then dust. Zoom to 100 per cent and walk the whole jar, top to bottom. Add a **new empty layer**, tick **Sample All Layers**, and heal onto that layer — every repair sits above the photograph and any one can be erased. Where a mark sits on the rim or the edge of the label, switch to the Clone Stamp and sample along the same side of that edge.

Do not remove a scratch that is actually on the product. That is the product.

Tone, colour, and out

- **Curves adjustment layer.** Black point where the histogram's data begins, then a very gentle S — upper quarter up two or three pixels of drag, lower quarter down the same.
- Check the label's white with the eyedropper. Around 240 to 248, not 255. A clipped label has lost its printing.

- **Is the honey the right amber?** Hold the actual jar beside the screen in daylight. That is the only test that matters, and it is what "colour not as pictured" on a return means. If it is off, a Hue/Saturation layer targeted at Yellows, masked to the honey alone.

Only now, and in this order:

1. Flatten a **copy**. Keep the layered master.
2. Resize to 2000 by 2000.
3. **Sharpen at this size**: Unsharp Mask, amount 70, radius 0.6 px, threshold 2. Or High Pass at 1 px on a duplicate set to Overlay, which can be masked off the background.
4. **Convert to Profile > sRGB.**
5. Export JPEG at quality 82. It should land between 250 and 500 KB. If it is 2 MB, the slider was at 100.
6. **Strip the metadata** — None in the export dialogue, or `exiftool -all= honey-front.jpg`. The GPS tag in a photo taken at home points at your house.
7. Name it for a human: `honey-500ml_front.jpg`.

What this is not

This gets you a clean, honest, sellable product photograph from a phone and a free editor. It does not get you a magazine cover, which needs controlled lights, a macro lens and years of practice.

The boundary is where lesson six put it. Straightening, dust, light and background are fair. Generating a lid that is not the lid, smoothing out a dent the buyer will receive, or making the honey a colour it is not — those change the claim, and the returns will say so anyway.

Today: do it once, on anything on your desk. Twelve minutes, and you have used every lesson in this course.

BEFORE YOU MOVE ON

A seller cuts a jar out onto pure white and buyers say the image looks fake, though the edge itself is clean and the colour is right. The most likely cause?

A. The cut-out needs more feathering; a harder edge than the rest of the photograph always reads as fake.

B. The contact shadow was removed along with the background, so nothing anchors the jar to a surface and it appears to hover.

C. The background should be 250 grey rather than pure white, and that would restore the sense of depth.

D. The image was sharpened before it was resized, which produced the artificial look.

CASE STUDY · MODULE 8

Fourteen seconds to the first picture, and the bill nobody had approved

A district education portal in Jaipur, where the admissions notice is read mostly on mid-range phones over mobile data.

The measurement

Somebody in the office opened the portal's admissions page in a browser with the network throttled to a slow connection, which is a thirty-second test that nobody had ever run. Fourteen seconds before any picture appeared. The page had thirty-one images on it and every one of them began downloading at once.

The causes were not subtle.

Almost every image was a **PNG**, because the staff who upload them were told years ago that PNG is the high-quality format. A PNG of a photograph is routinely three to six times the size of a visually identical JPEG, because the format is lossless and designed for flat colour and hard edges. The hero — a group photograph of a prize-giving — was 2.4 megabytes.

Almost every image was **3,000 to 4,000 pixels wide**, straight from a camera, displayed in a column about 700 CSS pixels across. On a ratio-2 phone that column needs roughly 800 real pixels. The browser downloads all four thousand of them, then scales down in encoded space with no gamma correction and no output sharpening, which is also why the pictures looked slightly soft.

And **nothing was lazy-loaded**, so a thumbnail in the footer competed for bandwidth with the photograph at the top of the page.

The choice

An image CDN. Ten minutes to switch on. One master per image and derivatives come from URL parameters — width, format, quality — so the widths and formats problem disappears without anyone re-exporting anything. It is a monthly bill against a budget line that does not exist, and it is a dependency:

every image on a government portal would then be served through a private company's infrastructure, on their pricing and their uptime.

Fix it properly, themselves. Re-export 900 images to three widths in three formats with a shell loop, write `srcset` and `sizes` into the article template, and change the upload rule in the CMS so staff cannot post a 4,000-pixel file at all. Free, permanent, and two days of a developer who is the only developer and does not have two days.

Do the cheap half. Fix the twelve pages that carry most of the traffic, set the loading and priority attributes site-wide, and leave the rest until there is time.

The temptation was the CDN, because it is a decision that can be made in a meeting and finished the same afternoon. The argument against it was not technical: it was that the underlying problem — staff uploading camera files — would be hidden rather than solved, and the bill would grow with the site.

Arguing in budgets rather than in settings

The meeting where this was decided nearly turned into an argument about the quality slider, which is an argument nobody wins because a quality number is not comparable between two encoders and does not describe an outcome at all. It scales a quantisation table. Feed it a plain studio backdrop and quality 82 gives a small, perfect file; feed it a crowd under autumn foliage and the same setting gives a large file that still shows artefacts.

What ended the argument was a budget. A full-width photograph on a page gets 100 to 300 kilobytes; a grid thumbnail gets 15 to 40; a whole article page including its images aims under a megabyte, and going over it is a decision somebody makes rather than something that happens.

That converts an aesthetic disagreement into arithmetic. If the hero has to be 1,000 pixels wide and fit 200 kilobytes, the quality setting is whatever achieves that, found by exporting two versions and flipping between them at 100 per cent rather than by nominating a number in advance.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

They did the cheap half first, in one afternoon. The twelve highest-traffic pages had their images re-exported: the 2.4-megabyte hero PNG became a 1,000-pixel JPEG at quality 80 and 180 kilobytes, converted to sRGB and stripped of metadata. Everything below the fold got `loading="lazy"`. Time to the first picture on the same throttled connection went from fourteen seconds to 3.1.

They got one thing wrong on the way, and it is the instructive part. The first pass marked every image lazy, including the hero, and the measured number got worse rather than better. The image at the top of the page is usually the element the browser's own load-time metric is timing, and marking it lazy pushes its request behind the others. Removing the attribute from the hero and adding `fetchpriority="high"` recovered more than a second.

The proper fix followed over the next month, one section at a time: an ImageMagick loop producing 400, 800 and 1,600-pixel versions in AVIF, WebP and JPEG, a `<picture>` block in the template with `sizes` written from the same definition as the stylesheet, and a CMS rule refusing uploads over 2,000 pixels with a message explaining why. The CDN was never bought.

One detail they nearly missed: several older PNGs carried an Adobe RGB profile, and the pages that had been converted rather than merely re-saved were the ones that stopped looking washed out on phones.

Why were the PNGs the wrong format here, and what did that cost specifically?

PNG is lossless and designed for hard edges, flat colour and transparency: screenshots, logos, diagrams. A photograph stored losslessly keeps detail that a viewer could never miss, so the file is routinely three to six times the size of a visually identical JPEG or WebP. On this page it meant a 2.4-megabyte hero where 180 kilobytes was indistinguishable, multiplied across thirty-one images, delivered to readers paying for mobile data. The correct rule is to pick the format by what is in the picture — photographs as JPEG, WebP or AVIF, and flat graphics and anything needing an alpha channel as PNG.

Marking every image lazy made the measurement worse. Why?

Lazy loading tells the browser not to fetch an image until it is near the viewport, which on a long page is the single biggest saving available — for everything below the fold. The image at the top of the page is usually the Largest Contentful Paint element, the thing whose arrival the browser's own performance metric is timing. Marking it lazy deprioritises exactly the request that defines when the page looks loaded, so the number goes up. The rule is everything below the fold lazy, the hero not lazy and marked high priority, and width and height on every image so the layout does not jump as they arrive.

The CDN would have solved the widths and formats in ten minutes. Why was fixing the dimensions first the better order?

Because dimensions are a first-order saving and compression is a second-order one: shaving bytes off a 4,000-pixel file being displayed at 800 is tidying a room that needs a wall moved. A CDN would have hidden that by generating correct derivatives on request, while staff carried on uploading camera files and the bill grew with the site and with the traffic. Doing the export themselves also let them change the cause — a CMS rule that refuses oversized uploads — which is the part that survives the next person to join the team. The CDN's real costs were the recurring bill against a budget line that did not exist and a dependency on a private service for every image on a government portal.

MODULE 8 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 An image sits in a 600 CSS-pixel column, and most readers are on phones with a device pixel ratio of 2 or 3. What width should you export?

A. 1,200 pixels
B. 1,800 pixels, to cover the ratio-3 phones
C. 600 pixels, and let the browser scale it up
D. 2,400 pixels, so it stays sharp if the layout ever widens

2 A photograph exported at 800 pixels looks crunchy, with pale rims around every branch. It was sharpened at 4,000 pixels and then resized. Why does that show?

A. Sharpening added detail that the resize could not preserve
B. The file was sharpened in linear light rather than on encoded values
C. The resize used bilinear instead of Lanczos
D. The halos were sized for the larger frame

3 A photograph edited in Adobe RGB looks flat and dull on a marketplace listing. What should have been done before export?

A. Assign the sRGB profile
B. Strip all metadata, including the profile
C. Convert to sRGB
D. Export at a higher JPEG quality so that less colour is lost

4 JPEG quality 82 produces a 95 KB file for a plain studio backdrop and a 470 KB file for a frame of autumn foliage. Why?

A. The foliage file was saved at a higher bit depth
B. The setting scales a table, it does not target a result
C. Foliage contains more saturated colour, which subsamples badly
D. The encoder switched to 4:4:4 automatically for the more detailed image

5 A three-second screen recording exports as a 4 MB GIF and as a 250 KB MP4. What makes GIF so expensive?

A. 256 colours per frame, dithered, with no motion compensation
B. GIF stores every frame losslessly at full colour depth
C. GIF cannot skip regions that are unchanged between frames
D. The frame rate is fixed at 30 frames per second in the GIF format

6 You mark every image on a long page as lazy-loaded and the measured load time gets worse. Why?

A. Lazy loading forces a second network request per image
B. Lazy images are decoded on the main thread and block rendering
C. The hero is what the metric times, and it was deprioritised
D. The browser cannot reserve space for a lazy image, so the layout shifts

EXAMINATION

Editing an Image — final examination

This paper covers the whole course: what an image file can and cannot survive, the non-destructive workshop, selections and edges, tone and colour, repair and retouching, compositing, the generative tools inside the editor, and the delivered file. Every question is about a mechanism — why a thing happens — rather than about the name of a menu item. You will be given 25 questions drawn at random from a larger pool, so no two papers are the same. The pass mark is 70 per cent. There is no timer: read at whatever pace suits you, in whatever language you think in, and come back to a question if it helps. If you do not pass, you can retake after thirty minutes and the paper will be drawn fresh, so there is nothing to be gained by memorising the questions. Passing opens the next course in the track. A teacher can open it for you at any time regardless, and that is recorded as an override rather than disguised as a pass.

On the website a sitting is 25 questions drawn from this pool and the pass mark is 70%. There is no time limit, there and here. Passing opens the next course in the track.

1 `1. What an image is made of`

A logo arrives from a motion designer and every edge in your document has a black halo. What is the likeliest cause?

A. Premultiplied data was imported as straight
B. It was cut from a dark background and the edge pixels are contaminated
C. The file is PNG-8, so its alpha is on or off
D. The layer's blend mode has been left on Multiply rather than Normal

2 `1. What an image is made of`

A logo with soft, anti-aliased edges was exported small to save bytes and now has visible staircase edges. What happened?

A. The export resampled it with nearest neighbour
B. It was saved as PNG-8, whose alpha is on or off
C. The soft edge was lost when the profile was converted to sRGB
D. The file was matted against a background colour before export

3 `1. What an image is made of`

You flatten a carefully worked 16-bit file to 8-bit for delivery and a band appears in the sky. What is the appropriate fix at that point?

A. Convert the delivered file back to 16 bits per channel
B. Raise the JPEG quality until the band disappears
C. Add one to two per cent noise, masked to the sky
D. Apply a Gaussian blur across the whole of the sky region

4 `1. What an image is made of`

A soft proof to sRGB shows a large saturated orange area collapsing into a flat patch. Which rendering intent addresses that, and what does it cost?

A. Relative colorimetric, which leaves in-gamut colours exactly where they are
B. Absolute colorimetric, which additionally reproduces the source white point
C. Saturation, which keeps the colours vivid at the expense of accuracy
D. Perceptual, which compresses everything slightly

5 `1. What an image is made of`

What does calibrating your monitor actually promise?

A. Repeatable judgements, and the screen ruled out as a suspect
B. That your screen and your client's screen will show the same colours
C. That an image will look the same on a phone as it does on your desk
D. That colours on paper will match what you saw while you were editing

6 `1. What an image is made of`

Purple fringing on out-of-focus highlights does not respond to the lens profile's chromatic aberration correction, although fringes in the corners of the same frame vanish. Why?

A. The profile was measured at a different aperture
B. The channels are differently blurred, not displaced
C. The fringe is a JPEG artefact rather than an optical one
D. Longitudinal aberration only appears after the geometry has been corrected

7 `1. What an image is made of`

Why do distortion and vignetting corrections belong before cropping and retouching rather than after?

A. They are only available while the file is still raw
B. Cropping locks the lens profile to the visible area
C. They resample every pixel in the frame
D. Retouching raises the noise floor and confuses the profile

8 `1. What an image is made of`

Why does a 24-megapixel sensor not record 24 million red, green and blue triples?

A. The file is compressed before it reaches the card
B. Two thirds of the photosites are reserved for autofocus measurement
C. The sensor samples colour at half resolution to save bandwidth
D. Each photosite sees one colour through a filter array

9 `2. The non-destructive workshop`

You must show forty client designs on the same photographed t-shirt, each following the fabric's folds. Which arrangement makes that forty double-clicks rather than forty warps?

A. A smart object warped and masked once, with its contents replaced
B. Forty duplicated layers, each warped and masked separately
C. A single layer with forty saved alpha channels in the Channels panel
D. An action that records the warp and replays it on each pasted design

10 2. The non-destructive workshop

A PSD arrives from a colleague with a warning badge on one layer and the logo missing from the canvas. What went wrong?

A. The document was saved without Maximise Compatibility
B. The layer was a linked smart object and the asset did not travel
C. The logo layer was set to Fill 0 rather than Opacity 0
D. The smart filters on that layer have not re-rendered since the file moved

11 2. The non-destructive workshop

You have two saved masks in the Channels panel and want a mask of the region where both are white. What is the operation?

A. Screen them together
B. Add them together and clip the result
C. Multiply them together
D. Invert one of them and then multiply

12 2. The non-destructive workshop

You spend twenty minutes building a difficult selection, export the result as a WebP, close everything, and come back tomorrow. Where is the selection?

A. In the WebP's alpha channel, ready to be loaded again
B. Recoverable from the export dialogue's recent settings
C. In the History panel, which reopens with the document
D. Gone, unless it was saved into a PSD or TIFF

13 2. The non-destructive workshop

A Hue/Saturation layer clipped to a cut-out product suddenly starts affecting the whole frame. What changed underneath it?

A. The base layer was filled or rasterised edge to edge
B. The clipped layer's blend mode was reset to Normal
C. The group containing them was switched to Pass Through
D. The document was flattened and then undone, breaking the clip

14 2. The non-destructive workshop

You want a floating drop shadow with the shape that casts it invisible. Which control does that?

A. Opacity at 0
B. Fill at 0
C. The layer's blend mode set to Screen
D. A layer mask filled with black over the shape

15 2. The non-destructive workshop

The efficiency indicator at the bottom of the window reads 55 per cent. What is it telling you, and what is the first thing to change?

A. The document is half its maximum size; nothing needs changing
B. Half the layers are hidden and still being composited
C. It is working mostly from disk; give it scratch space
D. The graphics card is being used for only half of the operations

16 3. Selections, cut-outs and edges

Your pen-tool path round a bottle is accurate to the pixel and the cut-out still reads as fake. The selection was loaded with no feather. Why does that matter?

A. Anti-aliasing has to be switched on before the path becomes a selection
B. The path was drawn on the edge rather than half a pixel inside it
C. Vector paths cannot describe the curvature of a bottle's shoulder
D. A real photographic edge is about half a pixel soft

17 3. Selections, cut-outs and edges

Quick Selection has given you eighty per cent of a difficult cut-out in two seconds and the remaining fifth is wrong. What is faster than tuning the tool?

A. Pressing Q and painting the rest with a soft brush
B. Raising the tolerance and re-running it over the failed area
C. Switching to Colour Range with localised colour clusters on
D. Running a segmentation model and intersecting the two results

18 3. Selections, cut-outs and edges

A wine glass photographed against white has to sit on a new background. Which method reproduces what glass actually does to light?

A. A carefully matted mask with decontaminated edge colour
B. A Multiply plate under and a Screen plate over
C. A mask plus a Gaussian blur to imitate refraction
D. A one-click segmentation mask with the interior opacity reduced

19 3. Selections, cut-outs and edges

A smoke plate photographed against black has to go over a night scene. What mask do you build for it?

A. A luminosity mask taken from the plate itself
B. A painted mask following the densest parts of the smoke
C. None — set the layer to Screen
D. A colour range selection on the black, inverted and feathered

20 3. Selections, cut-outs and edges

You select a red car with Colour Range and paint the selection flat yellow. The car goes dead and reads as a cut-out shape. What preserves the shading?

A. Feathering the selection before painting
B. Converting the selection to a path and filling that instead
C. Painting at 50 per cent opacity in two passes
D. A solid colour layer set to Colour blend mode

21 3. Selections, cut-outs and edges

A mask built from the blue channel of a high-ISO photograph separates hair from sky beautifully and has speckled edges. What is happening?

A. The blue channel is the noisiest, and the mask carries its noise
B. Blue clips first, so the sky has no detail left to mask against
C. Channel-derived masks are one-bit and cannot hold partial values
D. The Levels move you applied has introduced posterisation in the mask

22 3. Selections, cut-outs and edges

In a catalogue batch, why resize each product to fit 1,800 pixels before placing it on a 2,000-pixel canvas, rather than resizing straight to 2,000?

A. It keeps the resampling inside a whole-number ratio
B. It gives every product the same margin whatever its shape
C. It prevents the marketplace's own re-encoder from cropping the edges
D. It leaves room for the contact shadow to be added afterwards

23 4. Tone and colour

A flat, hazy, grey photograph needs rescuing. Which single move does the most, and why?

A. Raise contrast, which pushes the ends away from the middle
B. Add Dehaze, which estimates and subtracts the atmospheric veil
C. Drag the black point to where the data begins
D. Increase Vibrance, which lifts the muted colours more than the vivid ones

24 4. Tone and colour

The camera warns that a sky is blown, and the raw file turns out to hold a stop of it. Why is the camera wrong?

A. The camera meters the whole frame rather than the bright region
B. The warning is measured after the white balance multipliers are applied
C. The sensor's highlight headroom varies with temperature
D. The warning is computed from the embedded JPEG preview

25 4. Tone and colour

Dehaze pushed hard produces dark, oversaturated skies with cyan blotches over a white wall. What is failing?

A. The veil estimate assumes a dark channel everywhere
B. The local contrast radius is too large for the subject
C. The file is 8-bit and the correction has run out of levels
D. Saturation is being raised globally rather than by distance

26 4. Tone and colour

Dragging the Blues slider hard negative in a Black and White adjustment gives a dramatic sky and a grainy, blotchy one on a high-ISO frame. Why?

A. Monochrome conversion always doubles the visible noise
B. You are amplifying the noisiest channel
C. The adjustment works in 8-bit regardless of the document
D. Negative slider values invert the channel before weighting it

27 4. Tone and colour

The same LUT gives a good result on one frame and a muddy, contrastless one on the frame beside it. What is the explanation?

A. The LUT was built for log video and one frame is more log-like
B. The second frame is in a wider colour space than the first
C. A LUT maps by value, so exposure has to be normalised first
D. LUTs interpolate between entries and the second frame fell between them

28 4. Tone and colour

You are correcting skin and there is no colour checker and no reference photograph. Which check tells you most?

A. Whether the values match the published CMYK ratio for skin
B. Whether the red channel sits above the green and blue channels
C. Whether the lightness matches another portrait from the same shoot
D. Whether the hue is consistent across the face

29 4. Tone and colour

When matching forty frames from one setup, why adjust exposure per frame rather than reshaping the tone curve?

A. A curve reshaped per frame makes the set drift apart
B. Exposure is the only adjustment that can be synchronised
C. Curves are destructive and exposure is not
D. The curve has already been baked in by the raw developer

30 4. Tone and colour

Why is Vibrance the safer control than Saturation on a portrait against foliage?

A. It works in a perceptual space rather than on RGB values
B. It weights the increase against existing saturation
C. It applies only to the three secondary colour ranges
D. It clamps every channel below 250 so nothing can clip

31 5. Repair and retouching

Working retouchers finish a portrait retouch fully and then drop the whole group's opacity to 50 or 70 per cent. What does that correct for?

A. The loss of contrast that comes from stacking many local edits
B. The extra saturation that frequency separation introduces
C. Judgement drifting over forty minutes at high zoom
D. The difference between screen gamma and print gamma

32 5. Repair and retouching

Brightening the whites of the eyes towards white is the most recognisable amateur move in portrait work. What is the professional alternative?

A. Apply a small radial gradient of white at low opacity over each eye
B. Raise the contrast of the whole eye region on a masked curve
C. Paint a second catchlight to make the eye read as round
D. Desaturate the redness slightly

33 5. Repair and retouching

You ask content-aware fill to remove a person standing in front of a doorway and it fills the space with more wall. Why?

A. It copies patches from elsewhere in your own photograph
B. The sampling area excluded the doorway by default
C. Doorways are under-represented in its training data
D. The selection was not expanded enough to include the frame

34 5. Repair and retouching

What is the correct order for removing a parked van from a photograph of a street?

A. Fill the whole area first, then repair whatever comes back wrong
B. Rebuild the structural lines, then let the fill do the texture
C. Clone the entire region by hand and blend the edges afterwards
D. Generate the region, then paint the kerb and wall base over the result

35 5. Repair and retouching

Scanning a photograph out of a magazine produces coloured wave patterns across the image. Which fix gives the best quality?

A. Turn on the scanner software's descreen setting for magazines
B. Rotate the original a few degrees on the glass and rotate back later
C. Scan at 1,200 dpi and downsample to 300
D. Apply a Median filter of one to two pixels in the editor afterwards

36 5. Repair and retouching

Infrared dust removal works beautifully on colour negatives and turns a silver black-and-white negative into a smeared mess. Why?

A. Silver negatives are too dense for the infrared lamp to penetrate
B. The scanner cannot register the infrared channel against a monochrome frame
C. Black-and-white film has no dye layers for the algorithm to compare
D. The image itself blocks infrared and is read as damage

37 5. Repair and retouching

Why do family colour prints from the 1970s and 1980s share the same salmon-pink cast?

A. The cyan dye is the least stable and fades first
B. The paper base has yellowed with age and light
C. Colour negatives of that era had an orange mask that transferred
D. The prints were made on tungsten-balanced paper under daylight

38 6. Compositing: making two pictures one

Why does flipping a finished composite horizontally reveal errors you could not see a minute earlier?

A. Mirroring exposes asymmetries in the lens correction
B. Your eye has habituated and the mirror is unfamiliar
C. It reverses the direction of the light and so tests the shadows
D. It resamples the image and exaggerates any mask edges

39 6. Compositing: making two pictures one

An element is correctly scaled, lit and shadowed, and the composite still reads as pasted. Which single change usually helps most?

A. Add a rim light along the edge facing the key
B. Reduce the element's saturation slightly to match the plate
C. Let something in the scene pass in front of it
D. Add a second, fainter shadow to suggest a fill light

40 6. Compositing: making two pictures one

What is wrong with a uniformly blurred grey shape pasted under a composited object?

A. Grey is never the right hue for a shadow in daylight
B. A shadow should be built on a Screen layer rather than Multiply
C. It has no penumbra, so it cannot be blurred at all
D. A real shadow is darkest and sharpest at the contact

41 6. Compositing: making two pictures one

Why is a Multiply layer filled with a dark blue closer to correct than one filled with black, for a shadow in daylight?

A. The shadow is lit by the sky and by bounce
B. Blue survives JPEG compression better than neutral black
C. Multiply darkens neutral colours more than saturated ones
D. It compensates for the warm cast of the direct sunlight elsewhere

42 6. Compositing: making two pictures one

A focus-stacked macro photograph has a halo around a thin antenna. What produces it?

A. The frames were aligned without correcting for focus breathing
B. Its blurred version in other frames is wider than the sharp one
C. The contrast detection found the background sharper than the subject
D. The antenna moved slightly between the first and last frames

43 6. Compositing: making two pictures one

When blending two exposures of a room with a window, why mask the window with a rectangular selection rather than a luminosity mask?

A. Luminosity masks cannot reach pure white pixels
B. The two exposures differ by more than a luminosity mask can bridge
C. The window frame is a real hard edge
D. A rectangular selection can be feathered and a luminosity mask cannot

44 6. Compositing: making two pictures one

The element's darkest pixel measures 4 and the plate's darkest measures 22. What does that do, and what is the fix?

A. The element looks flatter; raise its contrast to match
B. Nothing visible; black point differences are below the detection threshold
C. The plate looks washed out; pull its black point down to 4
D. The element reads as separate; lift its blacks to the plate's

45 7. AI inside the image editor

Asking for a hundred per cent wider canvas in one outpainting operation produces drifting walls and bent horizons. Why does extending in stages fix it?

A. Each pass has more real pixels to condition on
B. Each pass uses a different random seed and the errors average out
C. Staged passes run at a higher internal resolution
D. The model retains the previous result in its context between passes

46 7. AI inside the image editor

A small generative fill on a 45-megapixel file comes back visibly softer than the photograph around it. What causes that?

A. The mask feather blurred the generated region's edges inward
B. Diffusion happens at one-eighth scale and is decoded up
C. The model applies noise reduction as part of its output stage
D. The fill was generated in 8-bit and placed into a 16-bit document

47 7. AI inside the image editor

Several of the strongest learned denoisers work on the raw file and refuse JPEGs outright. What is the reason?

A. JPEGs are 8-bit and the models require 14-bit input
B. Demosaicing must happen after denoising or the result is invalid
C. Raw noise has known statistics; a JPEG's has been transformed
D. The licences for those models forbid processing compressed files

48 7. AI inside the image editor

You generate three variations of a fill and two of them are about seventy per cent right in different places. What should you do?

A. Regenerate with a longer, more specific prompt
B. Lower the denoising strength and generate three more
C. Take the best one and sharpen the region that failed
D. Mask the good parts of each onto a layer

49 7. AI inside the image editor

Why is running a 2x upscaling model twice worse than running a 4x model once?

A. The second pass builds on the first pass's inventions
B. Each pass introduces its own tiling seams at different positions
C. Two passes take longer than one and produce the same result
D. The intermediate file has to be written to disk and re-encoded

50 7. AI inside the image editor

An online service reports that a photograph is 87 per cent likely to be AI-generated. What weight does that carry?

A. Enough to decline the image, given how specific the figure is

B. A weak signal, not a finding

C. It is conclusive if the service also reports the model family

D. It is reliable for generated images and unreliable for real ones

51 7. AI inside the image editor

The same photograph of a shop appears in a news story about the shop and on a poster advertising it. Why do the obligations differ?

A. Advertising is covered by different copyright rules

B. News photographs are published at higher resolution and show more

C. A record asserts the world looked like this

D. The poster has a named designer who takes responsibility for it

52 8. Out of the editor: size, sharpness and the delivered file

When is nearest neighbour the correct resampling filter rather than a lazy one?

A. When enlarging a photograph beyond about three times

B. For any image that will be re-encoded by a platform afterwards

C. When reducing a screenshot of text to fit a column

D. For an index map where a value means a category

53 8. Out of the editor: size, sharpness and the delivered file

Is resizing in ten per cent steps better than one pass, as the old advice says?

A. No, because repeated interpolation compounds error

B. Yes, because each step preserves more of the original detail

C. Yes, but only when enlarging rather than reducing

D. It makes no measurable difference with any modern filter

54 8. Out of the editor: size, sharpness and the delivered file

AVIF's characteristic failure at aggressive settings differs from JPEG's. What should you check before committing a catalogue to it?

A. Whether the flat areas have developed visible 8 by 8 blocks
B. Whether a texture-heavy region has quietly been smoothed
C. Whether saturated reds have acquired a fringe at their edges
D. Whether the file opens correctly in an older desktop image viewer

55 8. Out of the editor: size, sharpness and the delivered file

What does a lossless JPEG XL transcode of an existing JPEG give you?

A. A file with the compression artefacts of the original removed
B. A visually identical file at roughly half the size
C. A file about a fifth smaller, reversible byte for byte
D. A file that every current browser will display without a fallback

56 8. Out of the editor: size, sharpness and the delivered file

You have written srcset with three widths and the browser still downloads the largest file on a phone. Which attribute is missing?

A. loading, which controls when the fetch begins
B. width and height, which reserve the space before it arrives
C. fetchpriority, which ranks the request against others
D. sizes, which states how wide the slot will be

57 8. Out of the editor: size, sharpness and the delivered file

Stripping metadata with exiftool -all= leaves the delivered file looking washed out on some machines. What was removed that should not have been?

A. The embedded ICC colour profile
B. The EXIF orientation flag
C. The IPTC creator and copyright fields
D. The XMP block recording the editing history

58 8. Out of the editor: size, sharpness and the delivered file

A platform will re-compress whatever you upload. Why upload a large, high-quality file rather than one you have already squeezed to 90 KB?

A. Larger uploads are given a higher quality tier by most platforms
B. Their encoder should not be handed artefacts to compress
C. A larger file survives more re-uploads before it degrades
D. The platform keeps the original and serves it to logged-in viewers

59 8. Out of the editor: size, sharpness and the delivered file

A finished product edit is ready to leave the editor. What is the fixed order of the last four steps?

A. Sharpen, resize, convert to sRGB, export
B. Convert to sRGB, sharpen, resize, export
C. Resize, sharpen, convert to sRGB, export
D. Resize, convert to sRGB, export, then sharpen the exported file

Answers

Each entry gives the correct option, then what each wrong one reveals about how somebody was thinking. The second part is worth more than the first: a wrong answer here is a named misreading, not a mistake.

Lesson checks

1. The photo you were sent is too small, and nothing fixes that — C

A — Treats invention as an intensity dial rather than as what the model does at any setting.

B — Blames the container for information that was never in the file to begin with.

D — Confuses the print-size metadata field with the actual pixel count.

2. Three numbers per pixel, and why green counts most — C

A — Treats a mapping as damage: the conversion discarded a distinction, and re-saturating a grey pixel produces nothing, because hue information is gone once three channels are made equal.

B — Assumes the numbers were similar when the point is that the weighting differs: a properly weighted conversion separates them precisely because green counts 0.71 and red 0.21.

D — Reaches for gamut when the issue is the greyscale weighting; a wider space changes which colours are representable, not how hue maps to a single grey value.

3. Why the sky turns into stripes — C

A — Over-corrects into fatalism: depth does not recover lost levels, but it does change how subsequent rounding accumulates, which is a real difference.

B — Invents an interpolation step: conversion rescales existing values onto a wider axis and creates no intermediate tones between them.

D — Confuses the numeric size of a gap with its visible size: the mapping is proportional, so the displayed tones are unchanged by the conversion itself.

4. Mid-grey is not half the light — B

A — Blames kernel width for a space error: every kernel averages the same encoded values, so all of them darken the mix in the same direction.

C — Mistakes a numeric coincidence for a rule: zero has no special weight in an average, and the same darkening happens between any two very different values.

D — Reaches for profiles when the problem occurs identically within a single correctly tagged sRGB file.

5. Transparency is a number, and it carries a colour with it — B

A — Assumes a format limitation the question rules out: the alpha is described as clean and accurate, and 1-bit alpha would show staircase edges rather than a wrong interior colour.

C — Names a real bug with the wrong symptom: a premultiply mismatch darkens or brightens edges uniformly rather than preserving a background inside a transparent object.

D — Applies an edge fix to an interior problem: decontamination corrects thin partial-alpha boundaries, while the body of transparent glass legitimately contains transmitted colour that must be re-computed against the new backdrop.

6. Two thirds of the colour in your photo was calculated — C

A — Invents dual-exposure capture, which only a few specialist sensors do and which is not what standard highlight recovery uses.

B — Confuses precision with range: clipping is clipping at any depth, and extra bits subdivide the same captured range more finely rather than extending it.

D — Blames a real processing step for the wrong loss: noise reduction affects fine variation, not the fact that a channel has already been driven to its maximum value.

7. Eight by eight, and what the encoder threw away — B

A — Names a real artefact that does respond to quality: ringing is caused by coarse quantisation of high-frequency coefficients and does visibly reduce at quality 95.

C — Invents a per-colour rule: the encoder works on luminance and two colour-difference channels, and saturated edges in any hue behave the same way.

D — Offers a real phenomenon that does not match the evidence: generation loss shows as blocking and crawling edges across the whole image, not a fringe specific to saturated type.

8. Some colours have no number — A

B — Inverts the two intents: relative colorimetric preserves in-gamut colours exactly and clips the rest, while perceptual is the one that shifts everything inwards.

C — Reaches for bit depth when the colours in question have no sRGB representation at any precision: more levels subdivide a range that still excludes them.

D — Names a real setting with the wrong effect: black point compensation governs how shadows map, not whether saturated in-between colours survive the gamut boundary.

9. The monitor is a measuring instrument nobody calibrated — C

A — Confuses colour description with luminance level: an ICC profile characterises colour behaviour and does not change how bright the panel is set or how the editor judges exposure.

B — Shifts the problem outward when the editor is the one systematic outlier: several independent viewers agreeing is evidence about the source, not the audience.

D — Gets the adaptation backwards: eyes in a dark room become more sensitive to shadows, and the room is only part of the problem while the panel's absolute level is the main one.

10. Straight lines that are not straight, and colour that is not there — B

A — Assumes a spatial restriction that does not exist: profiles describe the whole frame, and lateral correction genuinely does nothing in the centre because the displacement there is zero.

C — Attributes it to the sensor: blooming is a charge overflow around extreme highlights and does not produce the wavelength-specific colour that varies with aperture.

D — Treats a category difference as a metadata mismatch: aperture does change the amount of longitudinal fringing, but no lateral correction at any aperture addresses a focus difference.

11. Every edit should still be undoable next week — B

A — Attributes banding to the compression ratio rather than to tonal range destroyed before compression.

C — Mistakes a defect written into the pixel data for a viewing problem.

D — Assumes a lossless container can restore tonal range the first edit already discarded.

12. The eraser is the tool that costs you the second revision — A

B — Assumes a mask can reveal pixels rather than only reveal pixels that are still present.

C — Treats destruction of data as a brush setting rather than as the act itself.

D — Expects a session-only history stack to survive flattening, saving and closing the file.

13. Twenty-seven blend modes, six ideas — D

A — Assumes a channel-specific edit: the shift appears with a composite RGB curve, and Luminosity does not redistribute anything between channels.

B — Treats a channel-separation mode as an opacity control: the tonal effect is delivered at full strength, which is exactly why the trick is useful.

C — Blames display management for an effect that survives into the exported file and appears identically on any screen.

14. The same three edits in a different order give a different picture — A

B — Offers a genuinely better tool for blending, but healing sampled above the adjustments would go stale in exactly the same way when the curve changes.

C — Assumes Normal mode is the fault when Normal is the correct mode for a clone layer; the issue is where the pixels came from, not how they are blended.

D — Identifies the right structural fact and draws the wrong remedy: putting the curve on top would adjust the patches too, but the repair should sit under the tone so it adapts rather than being corrected twice.

15. Why an adjustment stopped working when you put it in a folder — B

A — Applies the opacity uniformly to the whole document, which is not what a group opacity does: layers outside the group are unaffected by it in any mode.

C — Invents data loss: moving a layer into a group carries its mask with it, and a lost mask would make an adjustment apply more widely rather than less.

D — Gets the order backwards: the contents composite first and the opacity applies to the finished group result, which is precisely why the group must isolate.

16. One file, forty mockups, one change — C

A — Picks the wrong shared element: a duplicated layer gets its own mask, and a mask cannot change what image the contents show.

B — Describes a rendering interaction rather than a data link: two independent smart objects in Multiply would show two different designs perfectly well.

D — Names a real mechanism that would produce the same symptom but is not what happened here: the behaviour occurs with embedded objects too, because the duplicate shares the embedded source.

17. The panel where selections go to live — B

A — Names a real weakness of the blue channel that produces speckled edges rather than a uniformly hard outline, and the described symptom is the hardness, not the speckle.

C — Invents a restriction: Levels works on an alpha channel exactly as on any greyscale image, which is what makes this technique possible at all.

D — Confuses colour with gradation: a greyscale mask represents softness perfectly well through its intermediate values, which is exactly what the aggressive Levels move eliminated.

18. The file you will need is the one from Tuesday — A

B — Credits sync clients with content awareness they do not have: they replicate whatever bytes the application writes, with no notion of layers being lost.

C — Assumes history persists across a close, which is exactly the property it does not have in any of these editors.

D — Imagines flatten as a view toggle: it composites the layers into one and discards them, which is why it is one of the few commands worth saving a version before.

19. Eleven shortcuts and a scratch disk — C

A — Treats a symptom as a hard limit: layer count raises memory demand, but a document that fits with adequate scratch space stays responsive at forty layers.

B — Pushes a setting past the point where it helps: starving the operating system of memory typically makes the whole machine swap and feel worse, not better.

D — Identifies a real consumer that lives mostly in scratch rather than RAM, and reducing it helps only once the underlying free-space problem is addressed.

20. Working on a laptop that cannot hold the job — C

A — Applies a desktop diagnosis to a tool that has no scratch disk: a browser tab cannot overflow onto disk the way a native application does.

B — Assumes server dependence that does not match the tool's behaviour: Photopea does the editing locally in the tab and continues working through a connection loss.

D — Picks a real cost driver as the whole explanation: isolated group compositing raises memory use, but the abrupt termination is the allocation ceiling rather than any single feature.

21. Selections are easy until you reach hair — D

A — Treats a colour contamination problem as a geometry problem, and the fix eats the hair it is meant to save.

B — Reaches for a capture fault to explain an artefact created during compositing.

C — Assumes softening an edge removes colour rather than spreading it further.

22. Two handles and a curve: making the pen tool obey — A

B — Invents an automatic behaviour: no editor applies a threshold-based smoothing to bezier paths, which are rendered exactly as specified at any anchor count.

C — Describes a rendering fallback that does not occur: each segment remains a cubic curve regardless of how many anchors sit along it.

D — Treats a geometry error as an edge-softness problem: a larger feather blurs the wobble rather than removing it, and costs the crisp product edge the path existed to provide.

23. Stop fighting the selection tools and paint it — D

A — Changes which region is overlaid without changing the overlay colour, so a red overlay still disappears against a red jacket wherever it lands.

B — Abandons a tool over a display setting: nothing about Quick Mask depends on subject colour once the overlay colour is changed.

C — Makes the overlay harder to see rather than easier, worsening exactly the problem of distinguishing overlay from subject.

24. Tolerance is a wall, fuzziness is a slope — B

A — Claims a sampling limit that does not exist: the tool samples any value including near-black, and returns a usable selection there.

C — Describes the mode correctly then blames the wrong property: specular highlights are precisely what Colour mode preserves well on a mid-toned object.

D — Names a real setting that improves which pixels are chosen without touching the reason a hue shift on near-black produces no colour.

25. Masks with no edge anywhere — B

A — Assumes the sky is brighter than the building when both are near-white: narrowing the band affects them together, since the mask ranks by value alone.

C — Offers a channel trick that does separate blue sky from a neutral building, but describes a different technique rather than explaining why the luminosity mask behaved as it did.

D — Misapplies stack order to a single-layer photograph, where there is no separate building layer for an adjustment to sit above or below.

26. What the one-click button is actually doing — C

A — Names a real limitation with the wrong consequence: low mask resolution softens edges, while solidly filled enclosed regions come from the model's learned shape prior rather than from sampling density.

B — Attributes the behaviour to an optional cleanup stage, when the closed silhouette is typically produced by the model itself and persists with post-processing off.

D — Describes a flood-fill wand, not a segmentation model, and would also fail on the open background around the wheel, which the tool handled correctly.

27. Three equations, seven unknowns: why hair is hard — A

B — Conflates coverage with colour: a perfect alpha still leaves partial pixels carrying mixed colour, which is a property of the capture rather than of the mask.

C — Borrows a true fact about perceived brightness and applies it where it has no bearing: erosion removes pixels regardless of their hue.

D — Proposes the second of the three standard failed fixes: feathering spreads contaminated colour across a wider band rather than replacing it.

28. The four subjects that are not opaque — B

A — Reproduces the shape at the cost of the mechanism: in Normal mode the smoke layer still replaces the background inside the mask, so thin smoke reads as grey haze rather than as light added to the scene.

C — Applies an object-recognition tool to something with no object boundary: the model outputs coverage, and smoke's appearance is variable-intensity emission rather than partial coverage.

D — Picks the correct mode for the opposite case: Multiply suits dark smoke shot on white, and on a black plate it would darken the street scene everywhere the smoke is not.

29. Four hundred cut-outs, and the twelve that are wrong — B

A — Blames the thumbnail's encoding rather than its background: the failure modes described are invisible on white at any quality or scale.

C — Misreads sampling: 400 is the whole population here, and the problem is what the check can reveal rather than how many items it covers.

D — Raises a real but different question about the output format, which the flattening step in the pipeline has already settled.

30. Brightness/Contrast is one knob for a problem with five — B

A — Assumes clipping existed at capture when the edit itself pushed those tones past white.

C — Reads a tonal range problem as a colour intensity problem.

D — Confuses a non-destructive habit with the tonal mechanism causing the clipping.

31. There is no correct histogram — B

A — Reaches for precision when the values in question are at the maximum: extra levels subdivide a range whose top the red channel has already reached.

C — Names a real cause of flat saturated colour that would also alter the on-screen appearance globally, rather than a loss confined to the brightest petals.

D — Stops at the tool rather than the reading: the warning is useful, and it still needs interpreting against whether the clipped area is specular or genuine surface detail.

32. Two light sources, one global slider — B

A — Picks the wrong axis: yellow sits on the temperature axis, and magenta tint would leave a pink cast rather than a neutral grey field.

C — Invokes gamut clipping, which flattens saturated colour into a single saturated tone rather than desaturating it towards grey.

D — Offers a plausible-sounding capture setting that would desaturate the whole frame uniformly rather than specifically neutralising the dominant hue.

33. Shaping light by hand, on a grey layer — C

A — Catches noise and brushwork texture but not broad low-contrast unevenness, which remains below the visible threshold at any zoom until the contrast is exaggerated.

B — Does exaggerate the effect, and changes the actual edit rather than the view, so it risks the retoucher correcting against the wrong strength.

D — Diagnoses banding from stretching rather than uneven brushwork, and a histogram reports value distribution with no information about where in the frame it varies.

34. Why Clarity makes halos, and what Dehaze is estimating — B

A — Describes a genuine consequence of pushing highlights, which would appear as flat blown areas rather than a band that tracks every branch outline.

C — Correctly identifies that these tools work on luminance, then attributes the band to a channel mismatch rather than to the neighbourhood comparison that creates it.

D — Diagnoses banding, which appears as stepped terraces across smooth gradients rather than as a single bright band hugging an edge.

35. Saturation is a slider that destroys detail quietly — C

A — Reaches for precision rather than range: extra levels subdivide a gamut the printed colours still fall outside.

B — Names a real print constraint and places it beyond the editor's control, when a soft proof with the paper profile would have shown the loss in advance.

D — Picks a genuinely better working space for print without addressing the outcome: the colour still has to be mapped into the press gamut at output, where the same clipping occurs.

36. Monochrome is a decision about which colours become which greys — C

A — Invents an inversion: a negative weight reduces that hue's contribution to the grey value rather than flipping the channel.

B — Borrows a true fact about capture and applies it to a post-capture remapping, which changes displayed lightness without altering the signal-to-noise ratio that was recorded.

D — Attributes a real quantisation symptom to an invented internal limit: the adjustment honours the document's bit depth, and banding from 8-bit shows as smooth terraces rather than blotchy grain.

37. A look is a mapping, and a LUT is a table of it — A

B — Invents a fixed pipeline position: the application order is under the editor's control, and the mismatch persists however the two operations are sequenced if exposure is not normalised first.

C — Cites a real property of interpolation whose error is far too small to explain a visible difference in look between two frames.

D — Identifies a genuine secondary effect of grading an underexposed file while missing the primary cause, which would produce the same flat result on a clean, noise-free dark frame.

38. There is no correct number for skin — C

A — Treats a space conversion as the discrepancy when the numbers were already being matched in CMYK and the mismatch is with the person rather than with the medium.

B — Dismisses the only ground truth available: display variation is real, and it does not license overriding a person's account of their own appearance.

D — Assumes the ratio is what neutral produces, when a correctly neutralised photograph of dark skin lands on quite different numbers by construction.

39. Forty photographs that have to look like one shoot — B

A — Invents a colour side effect: a composite RGB curve alters contrast and can shift saturation, but it does not undo a white balance that has already been set consistently.

C — Raises a workflow inconvenience rather than the reason the results differ: settings can be copied at any complexity, and the inconsistency would persist if they were.

D — Attributes a visible mismatch to rounding that would be far below the threshold of perception in a single adjustment per file.

40. Three tools that all remove a spot, and they are not interchangeable — C

A — Treats an edge-averaging problem as a mode setting inside the same blending tool.

B — Blames sampling area size for behaviour the tool performs identically at any resolution.

D — Reaches for invention when an exact-copy tool solves it without changing the record.

41. Splitting texture from tone — B

A — Identifies a contributing setting rather than the mechanism: even a hard-edged selection crossing the boundary produces the halo, because the two layers hold complementary edge information.

C — Names the correct blend mode and invents a contrast dependency: Linear Light recombines the two layers uniformly and does not amplify by local contrast.

D — Points at a real settings mismatch between 8-bit and 16-bit workflows, which would produce a global tonal shift rather than a halo confined to one boundary.

42. Keep what is permanent, remove what is passing — B

A — Diagnoses a masking failure, which would produce a visible pale patch around the feature rather than the specific artificiality of the feature itself.

C — Names a real consequence of pushing brightness far enough, which is a later stage of the same error rather than the reason a merely brightened eye already looks wrong.

D — Proposes compounding the error: the catchlight is a specular reflection that is already the brightest correct element, and raising it further exaggerates the artificiality.

43. The product has to be the product — B

A — Invents a directional property of noise: random grain looks the same mirrored, and would not produce a visible mismatch at normal viewing size.

C — Describes a real execution fault that produces a localised seam rather than the overall wrongness of an object lit from both sides.

D — Raises an ethical frame where none applies: reproducing the true symmetrical shape of a symmetrical object misrepresents nothing, and the observed problem is visual.

44. A reflection is a photograph of the light source — B

A — Assumes a second optimum exists: rotating further changes which polarisation is blocked, and no angle suppresses a reflection that was never polarised.

C — Invents a depth restriction: the filter acts on all light reaching the lens regardless of how many surfaces it passed through.

D — Treats the filter as a neutral density reduction rather than a polarisation selector, and the exposure is compensated so relative brightness is unchanged.

45. Rebuild the structure first, then the texture — C

A — Assumes the doorway exists elsewhere to be found: widening the search helps when a match exists, and here the occluded content appears nowhere in the frame.

B — Describes a genuine limitation of some generative implementations rather than of patch synthesis, which copies full-resolution patches and preserves fine detail perfectly well.

D — Ascribes a preference for simplicity: the algorithm optimises patch similarity to the surroundings, and will happily reproduce highly detailed texture when similar detail exists nearby.

46. Moving pixels, and the straight line that gives it away — B

A — Names a real side effect of warping that shows only at high zoom on large moves, and would not be an immediate giveaway at normal viewing size.

C — Describes a genuine artefact of heavy stretching that requires close inspection to see, rather than something a reviewer identifies immediately.

D — Relies on a judgement about anatomy that varies with the viewer and the degree of change, where the background distortion is objective evidence visible in the file.

47. Getting the photograph off the paper — B

A — Misapplies a defect-detection channel: infrared identifies physical dust and scratches, and printed halftone dots are part of the image and fully visible to it.

C — Moves in the wrong direction: sampling more coarsely than the dot grid worsens aliasing and discards real image detail that upscaling cannot restore.

D — Adds precision to a spatial sampling problem: the waves come from grid interference and appear identically at any bit depth.

48. Tears, mould and the cyan that faded first — B

A — Offers a real precision improvement that subdivides the surviving signal more finely without adding colour information that the print no longer carries.

C — Treats an RGB file as though it had a cyan channel: reducing red is precisely how cyan is added in RGB, so the move described was correct.

D — Invents a property of channel curves: a single-channel curve changes colour balance and can raise saturation as easily as lower it.

49. Four losses that no tool undoes — C

A — Invents a colour restriction: deconvolution is applied per channel routinely, and many astronomical images are composed from multiple filtered exposures.

B — Substitutes precision for knowledge of the kernel: extra bits reduce quantisation noise without telling the algorithm what blur to invert.

D — Describes a real advantage of a stable mount without identifying the decisive difference, since a spatially uniform but unknown kernel is still ill-posed to invert.

50. Six mismatches, and the eye finds them all — B

A — Expects edge detail to survive a blur that is specifically chosen to destroy it: a 20-pixel blur removes any edge signature, which is why the test isolates broad tone.

C — Identifies a real mismatch whose effect a blur removes rather than reveals, since averaging grain produces the same mean value in both regions.

D — Names an error the test does not isolate: scale is judged from relationships to other objects, which the blur obscures rather than clarifies.

51. The horizon tells you where the camera stood — C

A — Substitutes proportion for position: two people of similar height standing on the same plane share an absolute crossing point on the body, not a matched fraction of their frame size.

B — Confuses the base's position with the crossing point: a distant figure's feet rise towards the horizon while the horizon still crosses their body at the same height.

D — Gets the grounding rule right and treats it as sufficient: correct contact fixes depth, and scale still has to be set independently or the figure reads as the wrong size at that depth.

52. Relighting a cut-out so it belongs — C

A — Treats density as the grounding cue: a uniformly darker shadow still lacks the tight occlusion at the contact point and reads as a hole rather than a join.

B — Applies a physical half-truth: shadows receive ambient and bounced light, so a black shadow looks more artificial rather than better grounded.

D — Names a real mismatch that would make the shadow look wrong in a different way, while an object with a correctly angled cast shadow and no contact shadow still floats.

53. Constructing what the light would have done — B

A — Increases density uniformly, which moves the shadow towards the black-hole look rather than towards the graduated falloff a real shadow has.

C — Names a genuine improvement to the shadow's colour that addresses a smaller cue than the uniform softness, which is what makes the shape read as a pasted silhouette.

D — Offers a refinement that matters on cobbles or grass and does little on the smooth surface where a uniformly blurred shadow most obviously fails.

54. A new sky changes the light in the whole photograph — B

A — Names a real and commonly missed refinement whose absence is subtle, where a scene with no shadows under a hard sun is an immediate contradiction.

C — Identifies a genuine error that would show as oddly-shaped clouds rather than as a scene whose lighting does not add up.

D — Describes a refinement of the spill technique: applying the grade too broadly is a subtlety, not the reason a shadowless building under a hard sun looks wrong.

55. Two exposures of one scene, and how not to make it look like HDR — C

A — Names a real bracket problem that produces doubled or ghosted edges rather than symmetrical bright bands, and does not explain the global flattening.

B — Identifies a genuine bracketing mistake whose symptom is mismatched sharpness across the frame, not halos tracking every high-contrast boundary.

D — Diagnoses quantisation, which appears as stepped terracing across smooth gradients rather than as bands hugging every edge.

56. Many frames, one view, and the pivot point that matters — B

A — Names a real requirement whose failure produces misalignment everywhere in the seam rather than a clean distant match with a broken foreground.

C — Attributes a doubling to a projection choice, which alters geometry smoothly across the image and never duplicates an object.

D — Describes a genuine bracketing hazard whose symptom is banding in tone, not a geometric doubling of a near object.

57. When depth of field is not enough — C

A — Names a real stacking error whose symptom is a zone of softness at a particular depth rather than a halo tracing every thin object's outline.

B — Identifies a genuine requirement whose failure produces progressive misregistration across the whole frame, worst at the edges rather than on the thinnest subjects.

D — Describes a genuine settings mistake that would produce a muddled, low-contrast composite rather than a sharp image with a consistent halo on thin structures.

58. The last tell: texture and depth — B

A — Names a real matching parameter whose mismatch is subtle, where the presence of two distinct grain fields is structural regardless of their colour.

C — Invents a stacking rule: grain applied above an element is exactly how the technique works, and the layer's position relative to one element is not the issue.

D — Treats the problem as a quantity to be tuned, when adding more of a separately-applied grain field makes the boundary between the two textures more visible rather than less.

59. Noise in, picture out: inpainting explained — C

A — Addresses how much of the original survives inside the mask, which does not change how little surrounding context the model has to match its edges against.

B — Steers what is generated rather than how it joins: a well-described fill drawn tightly still meets the surroundings with almost no overlap to reconcile.

D — Names a real cause of softness inside the filled area, which is a different artefact from a discontinuity at the boundary.

60. Getting a usable result in three attempts instead of thirty — B

A — Invents a resource trade: the compute cost is essentially unchanged, and the difference lies in what the generation is being steered towards.

C — Claims a functional restriction that does not exist: prompts work for removals and simply bias the result, which is the problem being described.

D — Assumes more description converges on this specific lawn, when the surrounding context already contains that information and a prompt competes with it.

61. Making a photograph bigger than it was taken — C

A — Preserves accuracy at the cost of the brief: in at least three of the five ratios the composition collapses, which is the problem the task set out to solve.

B — Produces five independent generations of the same background, which will differ from one another and give the set an inconsistent look across formats.

D — Names a genuinely useful seam-carving tool that works on sky and texture and visibly distorts any background containing structure across five different stretch factors.

62. Which upscaler, for which material — C

A — Compounds the problem: a second pass works from the first pass's invented detail, and the noise would be enlarged twice rather than once.

B — Fixes a different artefact: inconsistent tiling produces a regular grid of faint seams rather than texture distributed across the shadows.

D — Picks a genuinely more faithful family, which would render the noise more faithfully rather than removing the reason the result is unusable.

63. Reconstruction sold as noise reduction — B

A — Assumes compression cleans the file: JPEG discards high-frequency detail including some noise, and heavily noisy JPEGs remain visibly noisy after encoding.

C — Attributes the gap to bit depth, which limits how finely values can be represented rather than whether the noise matches the trained distribution.

D — Identifies a real ordering constraint and gives it the wrong reason, since the barrier is the changed noise characteristics rather than ownership of the demosaic step.

64. A guess about what colour the dress was — C

A — Blames sampling density for a pattern that affects large objects such as whole coats and shopfronts as much as small ones.

B — Invents a residual colour signal: a greyscale negative records no chrominance for any subject, natural or manufactured.

D — Correctly identifies that one model commits more strongly to saturated choices, which makes it confidently wrong more often rather than accurate.

65. On your own machine, with no graphics card — B

A — Mistakes free and self-directed for local: the image is uploaded to Google's machines, which is the exact disclosure the agreement covers.

C — Treats deletion as undoing transmission: the file has already been sent and may have been retained or used under the service's terms.

D — Assumes partial obscuring satisfies a confidentiality obligation, when the remaining material is still the client's unreleased photograph.

66. What can be proved about where an image came from — B

A — Assumes universal adoption: most cameras and most editors write no manifest at all, so absence is the ordinary case rather than evidence of removal.

C — Defers to a tool whose published accuracy is too low and too variable to settle anything, producing both false positives on processed photographs and false negatives on edited generations.

D — Describes a narrower scope than the specification has: a manifest records capture and conventional editing steps as well as generative ones.

67. The decision, and the parts nobody has settled — C

A — Makes quality the test: a more convincing fabrication is more misleading rather than more acceptable, which inverts the reasoning.

B — Draws the line at the tool rather than the act: wire services prohibit adding or removing content regardless of which technique performs it.

D — Treats client authorisation as sufficient, when a client cannot authorise fabricating a news photograph or an evidential image on anyone's behalf.

68. Nearest, bicubic, Lanczos: what the resize dialogue is computing — A

B — Reads the sharpening overshoot as a global tone change, when it is symmetrical about each edge and does not shift the average level.

C — Treats resampling as selection rather than averaging, which would make the result random rather than reliably too dark.

D — Confuses a profile conversion, which preserves appearance, with the arithmetic used during the resize itself.

69. The width to export, and why the browser will not fix it for you — B

A — Treats browser downscaling as a quality setting rather than as a resample, and implies a hardware limit that does not work that way.

C — Reverses the effect: downscaling tends to average compression artefacts away rather than amplify them.

D — Names a real problem with a different symptom — missing dimensions cause layout shift, not softness.

70. Sharpening adds no detail, and it goes last — A

B — Blames the resampling algorithm for artefacts that were written into the pixels before it ran.

C — Attributes sharpening halos to compression blocking, which looks different and has a different cause.

D — Inverts the correct order, and misidentifies a scale problem as an ordering problem between denoise and sharpen.

71. Why your image looks washed out on somebody else's phone — D

A — Attributes a gamut mapping shift to compression artefacts, which look different and affect detail rather than saturation.

B — Confuses the display's gamut with how the file's numbers are encoded and interpreted.

C — Treats a mis-tagging problem as a matter of taste, which then over-saturates anywhere the profile is honoured.

72. Stop guessing the quality slider — C

A — Offers a plausible secondary cause but does not explain why one setting produces opposite outcomes on two clean sources.

B — Treats a per-image mismatch as a global offset, which would also quadruple the size of the files that were already perfect.

D — Reaches for a lossless format for exactly the photographic material JPEG was designed for, at several times the size.

73. The formats that replaced JPEG, and the one that might — C

A — Invokes a channel that a photograph on an opaque background does not have and that would not affect surface texture.

B — Misreads effort as a quality control: lower effort costs bytes at the same quality, not detail.

D — Attributes a luminance-detail loss to colour subsampling, which would show on saturated edges rather than on grey weave.

74. GIF is a 1987 format and you are probably still exporting one — C

A — Names a real encoding option that affects how soon playback starts, not whether autoplay is permitted.

B — Invents a codec licensing rule; H.264 is hardware-decoded on phones and plays fine when autoplay is allowed.

D — Describes a layout-shift problem, which affects how the page settles rather than whether the video starts.

75. The fields that say who made it, and which platforms keep them — B

A — Attaches a rendering consequence to descriptive fields that no viewer uses for display decisions.

C — Confuses a geometry flag with colour interpretation; a missing orientation tag rotates an image, it does not desaturate it.

D — Assumes exiftool rewrites pixel data, when it edits the metadata segments and leaves the compressed image data alone.

76. One picture, six files, and letting the browser choose — A

B — Misreads the descriptor: `w` states the file's actual pixel width, which is precisely what lets the browser rank them.

C — Treats the two features as dependent, when srcset works on a plain img and picture adds format and art-direction choice.

D — Applies the right arithmetic to the wrong number: three times 340 is around 1020, which would still select the 800 or a 1024 file rather than 1600.

77. Your file is re-encoded on upload, and you can plan for it — B

A — Invents a policy; the pipeline applies the same settings regardless of input size.

C — Identifies a real step but the wrong one: the colleague's files are downscaled to the same width and still come out better.

D — Ties profile loss to the quality setting, when the platform strips metadata from every upload regardless.

78. One bottle, one window, twelve minutes — B

A — Applies an edge-softness fix to an image whose edge quality was already stated to be fine.

C — Borrows a rule about keeping product highlights visible and applies it to a problem about grounding the object.

D — Reaches for an output-order error to explain a compositing omission that has nothing to do with sharpening.

Module test — 1. What an image is made of

1. C

A resize asks for values at positions that were never sampled, so every classical filter computes a weighted guess between two pixels you already have. A guess between two known values cannot contain a detail that sat between them in the world and was never recorded. Lanczos is sharper than bicubic and still averages, and the ppi field changes only the print arithmetic, not a single pixel.

2. A

The Bayer colour filter array is half green, a quarter red and a quarter blue, so green is sampled twice as densely and carries the most measured detail and the least interpolation. The 0.7152 figure describes how much perceived brightness green contributes to your eye, which is a different fact and does nothing to noise. What JPEG halves is Cb and Cr, not a colour channel. Blue is usually the noisiest, and is still often the best channel for separating dark hair from a bright sky.

3. B

The document now has 65,536 available slots and still contains only 256 distinct values: the histogram is the same comb drawn on a wider axis. What the conversion buys is that everything you do from now on rounds into the finer grid instead of piling up on the coarse one, which is worth doing before a big tonal move on a JPEG. It is not the same as having captured 14-bit raw, and dither is still the defence against a band that survives to delivery.

4. D

Values in an sRGB file are perceptually spaced, roughly the 1/2.2 power of the light they stand for. Averaging two encoded numbers is not averaging light: a white and a black pixel average to 128, which emits about 22 per cent of full light, where the physically correct answer is about 188. Fine bright detail on a dark field is where the error is most visible, which is why linear-light resizing matters for starfields, city lights and white text on black.

5. A

Step two of the encoder stores Cb and Cr at half resolution in each direction, and red type on white is a hard edge in a channel that has just been halved. The quality slider scales the quantisation table at step four, which is downstream of the damage, so it cannot reach it. Forcing 4:4:4 helps; delivering the file as PNG or WebP is the real answer, because JPEG's assumptions are about photographs.

6. C

White balance multipliers differ per channel, so under any given light the three channels reach maximum at different exposures. When green is at the top and red and blue still carry values, the developer reconstructs the clipped channel from the two that survived, which is where the familiar one to two stops comes from. Where all three are at maximum there is nothing to reconstruct from and what gets painted in is a flat guess. The camera's warning is computed from the embedded JPEG preview, which is why it is pessimistic.

Module test — 2. The non-destructive workshop

1. D

Delete removes pixels. The shadow's values are not hidden somewhere, they are gone, and history is not saved with the document, so reopening the file gives you one state. A layer mask would have made this a soft white brushstroke of a few seconds, because the pixels would all still be sitting there under a stencil. A drop shadow style invents a shadow rather than restoring the one the camera recorded, which is a different picture.

2. B

Steepening a curve moves the three channels independently, and moving them apart by different amounts is an increase in saturation by definition, so the colour shift falls out of the arithmetic rather than being an extra thing the software did. Luminosity mode takes the tone from the adjusted layer and the hue and saturation from the base, so the contrast goes through and the colour stays. Lowering opacity reduces the contrast you wanted along with the shift.

3. C

A group's default mode is Pass Through, which renders as though the folder were not there, so an adjustment inside still reaches down through the group's floor. Giving the group any other blend mode, or an opacity below 100, composites its contents into a private buffer first and the adjustment can no longer see out. That is also the useful direction: to fence a grade to one part of a composite, put it in a group and set the group to Normal.

4. A

A repair copies pixels as they appear at the moment you paint. Painted at the top of the stack, the patch carries the tone the curve was producing that day, so moving the curve strands it. Put the clone layer directly above the background, beneath everything tonal, with Sample All Layers on, and the repair is adjusted by the same curve as the pixels around it. That is why the standard stack puts repair under tone.

5. D

Stamp visible writes a flattened snapshot onto a new layer and leaves the stack beneath intact. It does not update. Change anything below it and the document holds two versions of the picture, with the visible one stale, and an export that picks up the stamp exports the old one. Stamp late, name it with the date, and expect to delete and re-stamp after every revision.

6. B

Photopea lives inside a browser tab's memory allocation, roughly two to four gigabytes depending on the browser and the machine, with nowhere to spill when it runs out. GIMP and Krita have tile-based memory management and swap to disk, so they degrade gradually instead of stopping. That is an architecture difference rather than a quality one, and on a laptop short of memory, degrading gradually is usually the more useful failure.

Module test — 3. Selections, cut-outs and edges

1. B

The compositing equation says the recorded value equals alpha times the foreground plus one minus alpha times the background. That is three equations, one per channel, against seven unknowns: three for the foreground, three for the background and alpha itself. Infinitely many combinations produce the same recorded value, so every method has to supply information from elsewhere — a trimap, a known background, or a learned prior. It is mathematics, not a software limitation.

2. D

The alpha is correct; the colour underneath it is wrong, because the recorded value of a partial pixel genuinely contains the old background. Contracting deletes the strands, which are only two or three pixels wide. Feathering spreads the contamination further. Raising contrast destroys the partial values that made the edge believable. Decontamination recomputes the foreground colour from nearby known foreground while leaving the coverage alone, which is the only control aimed at the mechanism.

3. A

The wand's tolerance produces a yes-or-no answer, so every twig edge is either fully in or fully out and the boundary is a step you can only soften with a blanket feather afterwards. Fuzziness returns a partial value that falls off with distance from the sampled colour, so a pixel that is genuinely part sky and part branch is selected in proportion. A thousand twig edges is exactly that case, and it is why the wand is only really sane on flat graphics.

4. C

A luminosity mask sets each pixel's participation equal to its own brightness, so the adjustment fades continuously across the frame and there is nowhere for a transition to form. A halo is what happens when a treated region meets an untreated one along a line, and this mask has no line anywhere. The caveat is that the mask is a snapshot of the tones as they were, so it has to be rebuilt after any large tonal change beneath it.

5. B

A salient-object model was trained on human-drawn masks of mostly solid bodies, so it produces closed, plausible outlines rather than measuring contrast at a boundary. Enclosed gaps are the predictable casualty: handles, rings, mesh, the space between an arm and a torso. The internal resolution ceiling is a real and separate failure that shows as a soft, approximate outline, and thin structures vanish for a third reason again, because missing them barely costs the training objective anything.

6. D

A sample tells you whether something is wrong with the process, not whether every individual file is right. Twenty passing means there is no systematic fault; it says nothing about the one lamp whose lattice was chewed. That is why the sample comes after two cheap passes over the whole population — a thumbnail grid of the finished files, and a grid built from the masks on mid-grey — rather than instead of them.

Module test — 4. Tone and colour

1. A

Contrast is redistribution, not creation. Steepening the middle spreads the midtones over more output levels, and the only place that room can come from is the top and bottom of the curve, which flatten in exchange. That is why heavy contrast eats shadow and highlight detail, and why the useful habit is to look at the ends of your curve rather than only at the part you are dragging.

2. C

Luminance is weighted 0.2126, 0.7152 and 0.0722, so a red subject contributes little to it and can look perfectly exposed while the red channel is jammed against 255 and every petal has flattened. Only a per-channel view reveals that. The camera's histogram is computed from the embedded JPEG preview and is a different problem, and the combined RGB graph is not luminosity and is the least informative of the three.

3. D

White balance has two axes because real light sources are not all on the blackbody curve. Temperature runs blue to yellow; tint runs green to magenta. A fluorescent or cheap LED cast sits on the second axis, so no amount of the first slider reaches it. Two different sources in one frame is a separate problem again, and needs a second correction masked onto the minority light rather than any global control.

4. B

Mid-grey leaves the base unchanged in both Soft Light and Overlay, so the layer starts invisible; painting slightly lighter or darker grey lightens or darkens what is beneath without touching a pixel of it. Paint the opposite colour to undo locally, drop the layer opacity to reduce the whole effect, delete the layer to start again. The toolbar tools paint directly on the pixels and shift saturation as a side effect, which is why nobody who does this for a living uses them.

5. C

Shadows/Highlights, Clarity, Texture and unsharp masking are one operation at four different radii: blur a copy, compare each pixel with it, exaggerate the difference. Near a branch the blurred reference is a mixture of branch and sky, so a sky pixel is judged bright against it and brightened further, and a branch pixel is judged dark and darkened. The halo is the method rather than a defect, which is why less of it and a smaller radius are the cures.

6. A

Saturation pushes the channels further from their common average. The leading channel reaches maximum first, and every pixel past that point becomes the same value, so shading that ran across a petal collapses into one tone. Nothing warns you, and on a bright screen the flat version often looks more impressive, which is why the three-channel histogram is the check and Vibrance — weighted against how saturated a pixel already is — is the safer control on people.

Module test — 5. Repair and retouching

1. D

The healing brush borrows texture and then blends the patch's edges into the surrounding tone. Across a boundary the surrounding tone is both the pale wall and the dark worktop, so it averages them, and no setting stops that because averaging is the whole mechanism. Clone copies exactly, so sampling from the same side of the edge brings the correct tone with it. Heal the middle of the cloned patch afterwards if it needs settling.

2. A

The split puts colour and tone on the blurred low layer and detail on a high layer held around neutral grey. Healing matches texture and inherits local tone; on a layer that carries no tone there is nothing to inherit wrongly, so the repair is pure texture. Tonal unevenness is then handled separately on the low layer, where there is no texture to destroy — and never across a hard edge, where the blur has already mixed two regions together.

3. B

The test is whether the buyer receives what they saw. The dent is damage to this particular sample's box and the buyer's box will not have it, so removing it misleads nobody. The scratch is on the product as manufactured, so removing it shows something the buyer will not receive. Packaging arrives with the order and is part of what is bought, which is why the fourth option has it exactly backwards.

4. C

A warp is a displacement mesh and it moves everything within its influence, including whatever is behind the subject. A narrowed waist in front of a door frame leaves the frame bowed; a head slimmed against a window bends the mullion; tiled walls show curving grout. That is exactly what the freeze mask exists to prevent, and why warping the subject on its own layer is the safe practice. It is also how you check somebody else's work.

5. D

Once every channel is at maximum, every pixel in that region holds the same value and nothing distinguishes one from another, so there is nothing to recover. The other three are all faults in surviving data: a cast is a per-channel scaling, a crease is a tonal line over intact image either side of it, and distortion is a geometric transform with a measured profile. What can be done about a blown sky is replacement, which is compositing and should be disclosed if the picture is documentary.

6. B

Sharp or not sharp at 100 per cent is the fastest single test, because focus and motion blur are unrecoverable and everything else you might do depends on them. The ppi field is a print tag and changes nothing about the pixels. Running an upscaler first answers a different question and spends time you have not decided to spend. The full triage is five checks and five minutes, and it turns a bad week into a clear answer on day one.

Module test — 6. Compositing: making two pictures one

1. C

With a level camera the horizon sits at the camera's height, and every person standing on the same ground plane has it crossing them at the same point on their body regardless of how far away they are. That converts a guess about metres into a measurement you can take off both photographs. Focal length matters separately, because it governs the internal proportions of the element, and 24 mm against 135 mm does not survive any amount of scaling.

2. B

The contact shadow is caused by the object blocking ambient light arriving from every direction, so it exists even on an overcast day when nothing casts a directional shadow at all. It is dark, tight and relatively sharp, and it is the thing that puts an object on a surface. An object with a cast shadow and no contact shadow still floats; an object with a contact shadow and no cast shadow simply reads as a grey day. Build the contact shadow first.

3. D

The sky is the scene's light source, so its image appears in every reflective surface: glass, water, wet paving and car paint. Exposure, horizon placement and spill are all adjustments to values and can be graded. A grey sky sitting in forty balcony panes is content, and replacing it means masking and rebuilding each reflection separately, which is most of a day per photograph. The other unfixable case is a sky blown to pure white, where the edge pixels round fine branches carry no coverage information at all.

4. A

Compressing fifteen stops into a displayable range means cutting global contrast, which leaves the picture flat and grey, so the tone mapper restores contrast locally to compensate. Local contrast is the blurred-neighbourhood comparison from the tone module, and it produces halos at every strong edge by construction. Less of it and a smaller radius are the cures, and exposure fusion avoids the question entirely by never building a high-dynamic-range file in the first place.

5. C

Rotating about the tripod screw or about your wrists moves the lens through an arc, so near and far objects shift relative to each other between frames. The frames genuinely disagree about the scene's geometry and no stitcher can reconcile that, which is why the symptom is a doubled or sliced foreground with a perfect distance. Too little overlap, drifting exposure and the wrong projection are all real failures with different symptoms, and with nothing closer than about twenty metres, hand-held panoramas are fine.

6. D

Two grain patterns in one picture is the problem; one pattern across everything is the solution. Stamp the finished composite onto a layer, add a grey Overlay layer with monochromatic noise over all of it, and set the opacity until it matches the plate. The same argument governs vignetting, chromatic aberration and compression artefacts: unify them across the whole picture rather than correcting the pasted element on its own.

Module test — 7. AI inside the image editor

1. B

The unmasked pixels are re-imposed at every denoising step, so they pull the generated region towards agreeing with its edges continuously. That is a far stronger constraint than a text vector consulted once per step. Naming grass biases the result towards a generic idea of grass rather than towards this lawn, which is why removals go better with nothing typed while additions need the light and the material described.

2. C

The surrounding context is what keeps a generation consistent, and a mask drawn exactly at the object's edge leaves the model almost no overlap to condition on. Expanding the selection by 10 to 30 pixels and feathering 5 to 15 removes most seams, and it is the single most effective control in the whole feature. Working in several smaller passes helps for the same reason, because each pass has more context relative to its size.

3. A

An upscaler trained on reasonably clean inputs reads noise as texture and enlarges it into structured, wormy artefacts, so the noise has to go first and the model then has signal to work from. Sharpening exaggerates edges and creates halos, and enlarging afterwards enlarges the halos too, so it goes last and at the output size. The same order holds whether the enlargement is a model or a bicubic resize.

4. D

The model predicts chrominance from luminance using associations learned from modern colour photographs. Sky, foliage, skin, wood and brick have narrow colour distributions in the world, so the prediction is usually close. A dress could have been any colour, and a model trained with a squared-error objective hedges towards the mean of the possibilities, which is a muddy neutral. None of it is recovery: a greyscale file holds no colour information whatsoever.

5. B

Confidentiality is the argument that is not about money. A client's unreleased product, a photograph of a child, a medical image or anything under an agreement should not go to a service whose terms nobody has read and several of which reserve rights over uploads. Local models are usually a step behind the best hosted ones in quality. The licence still binds you, and disclosure is a separate obligation the machine does not discharge on your behalf.

6. A

Absence proves nothing, because a file with no manifest is the overwhelming default. A valid manifest is evidence of the chain a file went through rather than of the truth of what it shows; a broken one is a reason to ask questions. Platforms strip metadata when they re-encode, and a screenshot destroys it entirely, which is why the strongest provenance a working editor has remains the kept original plus a written delivery note.

Module test — 8. Out of the editor: size, sharpness and the delivered file

1. A

Export width is the slot's CSS width times the device pixel ratio, capped at 2 for photographs. A 3x file has 2.25 times the pixels and roughly 2.25 times the bytes for a gain at the edge of what anyone can detect at reading distance. Handing the browser something larger costs all of those bytes, gets downscaled without gamma correction, and throws away the sharpening you tuned for the delivered size.

2. D

Unsharp masking blurs a copy, subtracts it to find the edges and adds them back at strength, so it exaggerates transitions and adds no detail at all. The bright and dark rims it creates are as wide as the radius you chose, which was chosen for a 4,000-pixel frame. Resampling then stacks neighbouring halo pixels together and the small version crawls. Resize first, then sharpen at 0.5 to 1 pixel at the delivered size.

3. C

Assign changes the label without changing a single number, so the appearance shifts because the numbers now mean something else. Convert recalculates every number so the appearance survives into the new space. Software that ignores or strips a profile assumes sRGB, because everything assumes sRGB, and that is exactly what pulls a wide-gamut file's saturated colours inwards. Convert, embed, and accept that many platforms strip the profile anyway — which is harmless once the numbers are sRGB.

4. B

The quality number scales the quantisation table and does nothing else. A picture full of fine detail has far more coefficients competing to survive that table, so the file is larger and may still show artefacts, while a plain backdrop is small and visually perfect at the same setting. That is why one number across a set gives you files that vary wildly in weight and in quality, and why per-image targeting against a perceptual metric beats a fixed slider.

5. A

A palette of 256 colours forces dithering on anything photographic, and dithering replaces smooth areas that would have compressed to nearly nothing with high-frequency noise. There is also no motion compensation: a modern codec can say that this block moved eleven pixels left, while GIF can only skip unchanged regions, so a pan rewrites almost every pixel of every frame. Deliver MP4 or WebM with muted autoplay, or animated WebP where an image tag is required.

6. C

The image at the top of the page is usually the Largest Contentful Paint element, which is what the browser's own performance metric is timing. Marking it lazy pushes its request behind the others and delays the moment the page looks loaded. Everything below the fold gets the attribute; the hero gets a high fetch priority and no lazy attribute; and every image keeps its width and height so the page does not jump as the rest arrive.

Editing an Image — final examination

1. A *1. What an image is made of*

In premultiplied alpha the colour values have already been multiplied by the coverage. Treating that data as straight multiplies them a second time, so every partially transparent pixel goes darker and the outline reads as a black rim. The reverse mistake brightens edges instead. The fix is a checkbox in the import dialogue, not a mask. Contamination from an old background gives a coloured rim rather than a uniform black one, and PNG-8's one-bit alpha gives a staircase.

2. B *1. What an image is made of*

PNG-8 carries a palette and one entry in that palette means transparent, which is one-bit alpha: fully on or fully off. Soft edges have nowhere to live, so they become a staircase. PNG-32 carries 256 levels of alpha and WebP carries the same at a smaller size. Matting is a different thing again — it blends the semi-transparent pixels towards a chosen background colour, which looks right on that background and wrong on every other.

3. C *1. What an image is made of*

A band is a boundary between two adjacent levels with nothing in between. Adding a small amount of noise breaks that boundary into a scatter of pixels from each level and your eye averages them back into a smooth transition, which is exactly why film grain hides banding that a clean digital gradient shows. One to two per cent is enough and it survives JPEG better than the band does. Converting back up creates no levels, and blurring smears the picture without filling the gap.

4. D *1. What an image is made of*

Relative colorimetric keeps every in-gamut colour untouched and clips the out-of-gamut ones to the nearest point on the boundary, which is accurate for most of a picture and destroys detail in a large saturated area because several distinct colours land on the same edge point. Perceptual compresses the whole gamut inwards so the relationships between colours survive, at the cost of moving colours that would have fitted. Saturation is for charts, and absolute colorimetric is for proofing one press on another.

5. A *1. What an image is made of*

Nothing makes your screen match somebody else's, because their display is uncalibrated, at a brightness they chose, in a room you cannot see, possibly with a blue-light filter on. What calibration delivers is that the same file looks the same to you next month, that two machines in one studio agree, and that when a print comes back wrong the display is eliminated as a cause. There is one hard promise: with a calibrated screen and a paper profile, a soft proof predicts a print closely enough to approve it.

6. B *1. What an image is made of*

Lateral aberration magnifies the channels by slightly different amounts, so the fix is to rescale red and blue by a fraction of a per cent, which is why corner fringes correct cleanly and can even be derived from the image itself. Longitudinal aberration focuses the channels at different distances, so the channels sit on top of each other and simply have different amounts of blur. There is no geometric correction for that; the treatments are desaturating the fringe locally or accepting it.

7. C *1. What an image is made of*

Geometry corrections move the coordinates of everything, so anything already done is stretched along with the picture: a cloned patch that matched perfectly can end up smeared. Correcting also changes the frame's shape, so the corners need trimming, and if you have already cropped tight, that loss comes out of your composition. It is also why corner sharpness gets worse under heavy correction — existing pixels are being interpolated across a larger area.

8. D *1. What an image is made of*

A photosite counts photons and reports one number. Colour comes from a filter array over the sensor — half green, a quarter red, a quarter blue in a repeating tile — so every finished pixel has one measured channel and two calculated from its neighbours. That interpolation is demosaicing, and it is invisible on ordinary subjects and visible as moiré and false colour on fine repeating detail such as a distant brick wall or a tweed jacket. Free raw developers let you choose the algorithm, which is an improvement unavailable to anyone shooting JPEG.

9. A ***2. The non-destructive workshop***

A smart object stores the original and re-renders it through your transform and filters each time, so the warp, the blend mode, the mask and the shading are set up once. Double-click the thumbnail, paste the new design into the document that opens, save and close, and the mockup updates. Every downloadable mockup PSD works this way. An action replays a warp but cannot adapt it, and duplicated layers drift apart the first time anything changes.

10. B ***2. The non-destructive workshop***

A linked smart object stores a path rather than a copy, which is exactly what you want for an asset reused across a campaign — update it on disk and every document follows. It breaks the moment somebody moves the folder, renames the file, or sends the PSD without the assets, and the application asks you to relocate it. Embedded is the choice for anything leaving your machine, and File, Package collects a document and its linked assets into one folder.

11. C ***2. The non-destructive workshop***

Multiplying two masks intersects them: a pixel survives only where it is white in both, because anything times zero is zero. Screening them unions them, because black is invisible in Screen and either mask being white is enough. Those two facts are the whole of practical channel arithmetic, and Calculations is the dialogue that does it in one step, writing the result to a new alpha channel, a selection or a new document.

12. D ***2. The non-destructive workshop***

Alpha channels are per-document storage and they travel inside PSD and TIFF, adding negligible size and making a professional cut-out re-cuttable six months later. They are silently discarded on export to JPEG, PNG or WebP, and history dies when the document closes. A selection that took twenty minutes should be written into a channel before you press anything else, which is one command and the single cheapest habit in this module.

13. A ***2. The non-destructive workshop***

The bottom layer of a clipping chain is the base, and its transparency is what defines the shape everything above is confined to. Fill that layer edge to edge, or rasterise it onto an opaque background, and there is no transparency left to clip against, so the adjustments apply everywhere. Group blend modes govern whether adjustments inside a group reach the layers below, which is a related and separate mechanism.

14. B ***2. The non-destructive workshop***

Opacity fades everything about a layer including its layer styles, so at zero the shadow goes too. Fill fades the layer's own pixels and leaves the styles at full strength, which is exactly the floating-shadow case. It is a specialist control and knowing it exists mostly saves an argument with a file that appears to be behaving strangely. A black mask hides the shape and, in most implementations, the style with it.

15. C ***2. The non-destructive workshop***

Anything below 100 per cent means the application has run out of memory and is writing to the scratch disk; below about 60 it is mostly working from disk and you can feel every brushstroke. The fixes in order are to point the scratch disk at the fastest drive with the most free space, keep 20 to 60 gigabytes free there, hide the groups you are not working on, and stamp a checkpoint so the stack beneath it stops being re-composited.

16. D ***3. Selections, cut-outs and edges***

A zero-feather edge is aliased and reads as cut with scissors instantly; two pixels of feather reads as a soft glow and is fake for the opposite reason. On a normally shot photograph the subject's own edge is roughly half a pixel soft, so 0.3 to 0.5 pixels is what matches it. Sitting the path half a pixel inside the edge is a separate and also necessary discipline: on the edge leaves a rim of old background all the way round.

17. A ***3. Selections, cut-outs and edges***

Quick Mask displays the selection as a paintable greyscale image, so every painting tool becomes a selection tool: black removes area, white adds it, grey adds partial selection, and a gradient or a texture brush works too. Tuning a threshold changes the answer everywhere at once, which is why a tool that is eighty per cent right rarely becomes ninety-five per cent right by adjustment. Eight brushstrokes beat any amount of slider work, and they produce a better edge because you are deciding rather than thresholding.

18. B *3. Selections, cut-outs and edges*

Alpha is a single coverage number per pixel and it cannot filter colour, so a correct mask on a wine bottle still gets the transmitted colour wrong. Shot against white, the frame records what the glass removes, which goes underneath on Multiply; shot against black it records what the glass adds, which goes on top on Screen. The new background then genuinely passes through, darkened where the glass is thick and with the real highlights over it. The technique is forty years old and no mask matches it.

19. C *3. Selections, cut-outs and edges*

Smoke, steam, flame and spray are emission rather than coverage. Against black the background contributes nothing, so Screen makes the black vanish and the smoke adds exactly what it should, with a Levels layer on the plate to control how much shows. A mask would destroy the soft variable density that makes it look real, which is why every stock library sells smoke and bokeh shot against black. Dark smoke against a bright sky is the mirror case: shoot against white and Multiply.

20. D *3. Selections, cut-outs and edges*

A colour selection includes the highlights and shadows on the object, which are the same hue at different lightnesses. Painting a flat colour replaces all of that shading with one tone. Colour blend mode takes hue and saturation from the fill and luminosity from the photograph, so the shading survives untouched. A Hue/Saturation layer masked to the selection, shifting hue, does the same job. Neither works on a black car, which has no hue to shift and no shading in the colour channels.

21. A *3. Selections, cut-outs and edges*

The mask is a copy of a channel, so whatever is in that channel is in the mask, and in a high-ISO frame blue carries the most noise. A one or two pixel blur on the alpha channel removes the speckle at negligible cost to the edge, or you build from green and accept less separation. The related caution is not to push Levels so hard that the transition zone becomes fully black and white: push in the flat regions, and paint the interior solid by hand instead.

22. B ***3. Selections, cut-outs and edges***

A tall bottle and a wide tray resized to the same canvas dimension end up at different visual scales, and the grid looks wrong even though every file is identically sized. Fitting inside 1,800 and then centring on 2,000 gives each item the same hundred pixels of breathing space in whichever direction it is longest. That kind of decision — canvas, margin, background value, shadows on all or none, machine-readable names — is what makes a set look commissioned rather than assembled.

23. C ***4. Tone and colour***

A flat photograph is one whose data never reaches black, so the histogram has a gap at the left. Setting the black point where the data actually begins uses the range that exists and costs almost nothing, and it fixes most flat images on its own. Hold Alt or Option while dragging and the preview shows only the pixels you are about to destroy, so you can stop at the first specks. Contrast, Dehaze and Vibrance are all larger interventions applied before the cheap one.

24. D ***4. Tone and colour***

The histogram and the blinking highlight warning on the back of a camera are computed from the embedded JPEG preview, which already carries the camera's contrast curve, saturation and white balance. The raw data underneath has not had any of that applied, and typically holds between half a stop and a stop and a third more highlight than the display suggests. Setting a flat picture style brings the preview closer to the raw and costs nothing, because the raw is unaffected by it.

25. A ***4. Tone and colour***

Dehaze is not local contrast; it estimates atmospheric scattering, usually through the dark channel prior — the observation that in a haze-free outdoor photograph almost every small patch contains at least one very dark colour channel. Where the scene genuinely has no dark pixels, such as a white wall, snow, or fog that is the subject rather than an obstruction, the veil estimate is wrong and the subtraction produces blotches. That is also why negative Dehaze is a quick way to fake atmospheric distance.

26. B ***4. Tone and colour***

The six sliders decide how strongly each hue contributes to the resulting grey, so pushing one hard is a large weighting change applied to one channel — and blue is the channel carrying the most noise. On an 8-bit JPEG the same move also stretches 256 levels across a wider range of greys and bands. Monochrome conversion benefits from 16-bit and from starting in a raw developer more than almost any other operation, precisely because it leans so hard on individual channels.

27. C ***4. Tone and colour***

A lookup table says, for a given input colour, what the output should be. It knows nothing about the photograph. A frame that is a stop darker lands on a different part of the mapping and comes out wrong, which is the commonest complaint about LUT packs and always has the same answer: normalise exposure, contrast and white balance first, then apply the look. Applying it on an adjustment layer at 40 to 70 per cent, with the histogram open, catches the clipping a strong LUT can hide.

28. D ***4. Tone and colour***

Skin covers an enormous range of hues, lightnesses and saturations and there is no single set of numbers that is correct; the widely quoted CMYK rule was derived from light skin on coated stock and lightens and pinkens darker skin when applied as a target. What is genuinely diagnostic is consistency: skin should be one hue family across a face, varying in lightness and saturation. A hue split between forehead, cheek and jaw is a real fault, usually from mixed lighting or a heavy-handed adjustment.

29. A ***4. Tone and colour***

Copy the hero frame's whole history stack to the set and the frames start identical. Exposure shifts everything uniformly, so a frame that needed a third of a stop stays in the same family as its neighbours. Re-shaping the curve changes the relationship between tones, and doing that differently on forty frames reintroduces exactly the inconsistency the sync was supposed to remove — one small, reasonable decision at a time.

30. B *4. Tone and colour*

Vibrance applies a saturation increase inversely weighted by existing saturation, so already-vivid colours barely move and muted ones move a lot, and it carries a specific protection for the orange-yellow hues in the skin range. That is a weighting curve rather than any understanding of the picture, and it is why a boost greens the leaves without turning the face orange. Both controls have a smaller useful range than people expect: plus ten to plus twenty is visible, plus forty looks processed.

31. C *5. Repair and retouching*

After forty minutes at 200 per cent every remaining mark looks intolerable, and the version that looked right at minute forty looks processed the next morning. Reducing at the end corrects for that drift without redoing anything. The companion check is to toggle the before and after at fit-to-screen: if the change registers as a change rather than as a better photograph, it is too much. Most over-retouching happens at zoom levels nobody will ever view the picture at.

32. D *5. Repair and retouching*

The whites of eyes are a pale, slightly grey or warm tone with visible vessels, and pushing them towards white is instantly readable as retouching. A twenty to thirty per cent desaturation of the redness on a masked Hue/Saturation layer does what people are actually after. Then a slight lift of contrast on the iris and a gentle burn round its outer edge, which is naturally darker and makes the eye read as round. Leave the real catchlight exactly as it is; a painted one is obvious and removing one kills the eye.

33. A *5. Repair and retouching*

Patch-based synthesis searches the rest of your image for the best-matching patch and stitches it in, iterating so neighbouring patches agree. Everything it produces really was somewhere in your photograph, which is both its honesty and its limit: with no other doorway in the frame there is nothing to copy. It also has no model of geometry, so straight lines break and receding surfaces come back at the wrong scale. Diffusion inpainting would invent a doorway, which is a different operation with a different problem.

34. B *5. Repair and retouching*

Patch-based tools are good at texture and have no model of geometry, so the method reverses the usual order: identify the lines that must continue through the hole — the kerb, the wall base, the roof line — and copy clean sections into place first, transformed to the right angle. Then hand the fill only the flat texture job. Then unify the tone across the repair with a masked curve, which is the step most often skipped, and add grain back, because synthesised areas are cleaner than the photograph.

35. C *5. Repair and retouching*

The waves are interference between the scanner's sampling grid and the printing screen's grid of halftone dots — the same aliasing mechanism as demosaic moiré. Oversampling means many samples land inside each printed dot, so averaging them down removes the pattern properly rather than blurring over it. Descreen is a targeted blur and costs sharpness; rotating the original breaks the interference and is crude but free; a Median filter afterwards handles residual dots better than a blur does.

36. D *5. Repair and retouching*

The technique works because dust and scratches block infrared while the image dyes do not, so a fourth infrared channel maps the physical defects. In a traditional silver black-and-white negative — and in Kodachrome — the image is made of material that is itself opaque to infrared, so the whole picture is detected as damage and dutifully repaired away. Those two cases are the exceptions that catch everybody, and the answer is to switch the feature off and clean by hand.

37. A *5. Repair and retouching*

A chromogenic print holds cyan, magenta and yellow dye layers which do not fade at the same rate, and in the common processes the cyan goes first. Magenta and yellow together read as red-orange. The correction follows directly: add cyan back by pulling the red channel down in the midtones and highlights. The limit follows just as directly: where the cyan has gone entirely there is no information to amplify, so the neutralised result is low in saturation and pushing harder produces noise.

38. B ***6. Compositing: making two pictures one***

Visual habituation is real: an hour with a file makes you the worst available judge of it. A mirrored version is a new image to your visual system, so the inconsistencies that had stopped registering come back. It is one of three cheap checks that each find a different class of error — shrinking to 300 pixels keeps errors of tone and light direction while discarding edge detail, and a heavy blur leaves only the broad tonal structure, which is where an exposure mismatch shows.

39. C ***6. Compositing: making two pictures one***

Occlusion is unambiguous depth information: a table edge or a plant crossing in front of the element tells the viewer where it sits in a way no amount of light matching can. Real objects in a real scene are partly hidden by other objects, and an element floating with nothing overlapping it looks added even when everything measurable is right. A rim light is the next most valuable addition, and it is worse than nothing on the wrong side.

40. D ***6. Compositing: making two pictures one***

A real shadow has three zones: a dark umbra where the source is entirely blocked, a penumbra that widens with distance from the object, and ambient fill from the sky and from bounce. So it is dark and tight where the object touches and progressively softer and lighter as it recedes, and it takes the shape of whatever surface it falls on. The colour point is also true — a daylight shadow is bluish rather than black — but uniform density is the error that makes composites float.

41. A ***6. Compositing: making two pictures one***

A shadow is not an absence of light, it is a region the key light does not reach and everything else does. Outdoors that everything else is mostly skylight, which is genuinely blue, and bounce from nearby surfaces, which takes their colour — beside a red wall a shadow is reddish. Black is what a shadow would be with no ambient light at all, which happens nowhere outdoors. Density matters too: 20 to 40 per cent on Multiply reads as a shadow, 80 per cent reads as a hole.

42. B *6. Compositing: making two pictures one*

The algorithm picks each pixel from the frame where local contrast is highest. A thin structure that is sharp in one frame is a wider, softer smear in the others, so around the sharp core there are pixels whose best-contrast frame is one where the antenna was blurred. The result is a ring of doubled or blurred detail. More frames with smaller focus steps soften the transition, and hand-retouching from one source frame fixes what remains.

43. C *6. Compositing: making two pictures one*

A luminosity mask is the right tool where the transition in the scene is genuinely gradual, because it has no boundary anywhere and therefore produces no halo. A window in a wall is not gradual: there is a real hard edge at the frame, and blending softly across it puts a visible glow around the frame. Match the mask to the scene — soft where the scene is soft, hard where the scene has an edge — and the blend beats any tone-mapped merge.

44. D *6. Compositing: making two pictures one*

An element with deeper blacks than its surroundings has more contrast than the picture it is sitting in, and the eye reads that as a separate photograph even when everything else has been matched. Lifting the element's blacks to meet the plate's is quick, measurable with the eyedropper, and frequently the difference between a composite that works and one that does not. Pulling the plate down instead throws away shadow detail across the whole frame to fix one element.

45. A *7. AI inside the image editor*

The unmasked pixels are what constrain a generation, and they are re-imposed at every denoising step. Asking for a very large region from a thin strip of real image along one edge leaves the model conditioned on almost nothing, so it drifts towards whatever its training data suggests. Twenty to thirty per cent at a time, with a generous band of the existing image inside each generated region, keeps every pass strongly constrained by real pixels.

46. B ***7. AI inside the image editor***

Most systems compress the image into a latent grid eight times smaller in each direction, run the diffusion there, and expand the result back with a decoder. Fine detail is therefore decoded rather than generated, and the ceiling is set by that grid. The workaround is to copy a region around the fill into a new document at a size close to the model's native resolution, fill there, and composite the result back — more steps and considerably sharper.

47. C ***7. AI inside the image editor***

Noise in raw data is dominated by photon shot noise, which is Poisson-distributed and predictable from the signal level, plus a read-noise component characteristic of the sensor. A model trained on that distribution has a well-defined problem. Once the file has been demosaiced, white balanced, tone curved and compressed, the noise has been through several non-linear transformations and is correlated between neighbouring pixels, so it no longer matches what the model learned.

48. D ***7. AI inside the image editor***

A generation that is seventy per cent right is a useful layer, not a failure, and you are not obliged to accept a whole result. Put each on its own layer, mask it, and keep what worked. That single habit separates people who get good results from people who conclude the tool does not work. Longer prompts dilute the conditioning rather than compounding it, and the surroundings constrain an inpaint far more than any wording does.

49. A ***7. AI inside the image editor***

An upscaler predicts plausible detail consistent with the evidence it is given. After the first pass, much of the fine detail it is looking at is detail it produced itself, so the second pass compounds invention on top of invention with no new information about your photograph. Use the model trained for the factor you need. The related order rules are equally mechanical: denoise before upscaling, because noise is read as texture and enlarged, and sharpen only at the end.

50. B ***7. AI inside the image editor***

Detectors learn the artefacts of particular generators, so they degrade as generators change, they produce false positives on heavily processed real photographs and false negatives on generated images that have been edited or re-encoded. Published evaluations repeatedly find accuracy well below the marketing claims, and the error rates are not uniform across image types. Do not accuse anyone on the strength of one, and do not defend yourself with one.

51. C ***7. AI inside the image editor***

The file is identical; the use is not. A record makes a claim about the world, so adding or removing content changes what it asserts, which is why wire services and photojournalism competitions prohibit it and why careers have ended over a cloned sensor spot. A construction claims nothing beyond itself. Answering which one you are making, before opening any tool, resolves almost every hard case in this module — and writing down what you did covers the rest.

52. D ***8. Out of the editor: size, sharpness and the delivered file***

Nearest neighbour copies the closest pixel and never invents a value in between. That is correct for pixel art, where the blocks are the point, and for index maps — a mask where 1 means sleeve and 2 means collar, and an averaged 1.5 would be meaningless. Everywhere else it aliases fine detail into moiré when reducing and produces hard blocks when enlarging, and the resize dialogue's dropdown is not decoration.

53. A ***8. Out of the editor: size, sharpness and the delivered file***

It was real advice for one bicubic implementation in one application around 2001. With a modern Lanczos or the current bicubic, each pass is an interpolation of an interpolation, so the errors accumulate and the result is softer than a single pass. It is also a claim you can settle yourself in five minutes: one image, two routes, compared at 100 per cent, which is the right response to most advice of this kind.

54. B ***8. Out of the editor: size, sharpness and the delivered file***

JPEG fails by blocking and ringing, which look like damage and are therefore noticed. AVIF pushed hard smooths instead: fine grain, distant foliage and fabric weave quietly become flat, and the image looks clean rather than broken. That is the more insidious failure because nothing about it says a mistake was made. Open a texture-heavy region at 100 per cent and compare before converting several hundred files.

55. C ***8. Out of the editor: size, sharpness and the delivered file***

JPEG XL can re-encode an existing JPEG to roughly eighty per cent of its size and turn it back byte for byte, so an archive of a million JPEGs shrinks by a fifth with nothing lost and nothing re-encoded. It does not repair artefacts, because the data is being repacked rather than reinterpreted. Browser support is the genuinely unresolved part — Safari ships it, Chrome implemented and then removed it — so do not build a pipeline whose only output is JPEG XL today.

56. D ***8. Out of the editor: size, sharpness and the delivered file***

srcset lists the files and their real pixel widths, which is a statement of fact. sizes tells the browser how wide the slot will be in CSS pixels at each viewport width, and without it the browser assumes the image fills the viewport and picks accordingly. It is the attribute people leave out, and leaving it out breaks nothing visible while serving files several times larger than necessary. The others matter, and they are answering different questions.

57. A ***8. Out of the editor: size, sharpness and the delivered file***

A blanket removal takes the ICC profile with everything else, and a file with no profile is read as sRGB by every piece of software there is. That is harmless if you converted to sRGB first, and it is exactly the washed-out failure if the file was still in a wider space. Keep the colour space tags when stripping, bake any rotation into the pixels rather than trusting the orientation flag downstream, and write back the four fields worth keeping: creator, copyright, credit and description.

58. B ***8. Out of the editor: size, sharpness and the delivered file***

Your file is an input to their pipeline rather than the thing that gets served. Compression artefacts in what you send become the input to their compression and the two stack, so a pre-squeezed upload comes back worse than a generous one. It is counter-intuitive: you upload a larger file to get a smaller, better one back. The other levers are converting to sRGB, matching the width they actually serve, and sharpening slightly less than for final delivery.

59. C ***8. Out of the editor: size, sharpness and the delivered file***

Sharpening exaggerates the edges that exist at the scale it is applied, so it must come after the resize and at the delivered size, or the halos are the wrong width and the resampling makes them crawl. Converting to sRGB after that keeps the numbers right for everything downstream, and the export is last. The layered master is saved before any of it, and the metadata is stripped on the way out, which is where the GPS tag from a photograph taken at home would otherwise go.