MAKE THINGS

Editing an Image

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

First edition, September 2026

Addaly

Series editor: Dr Mustafa Mun

Draft, open for academic review

MAKE THINGS

Editing an Image

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

Series editor: Dr Mustafa Mun

Addaly

First edition, September 2026 · Draft, open for academic review

Editing an Image

First edition, September 2026, build 62a26075.

Written by AI models to a written standard, with Dr Mustafa Mun as series editor. Exam keys checked blind by a second model: 3,665 of 3,666 agreed. Academic review invited.

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.

How to cite: *Mun, M. (2026). Editing an Image. Addaly. addaly.com/learn/image-editing.*

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.

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. The answers to every check, test and examination question are at the back.

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

Brief contents

1

MODULE 1

What an image is made of

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.

Lesson 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:

Figure 1.1

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.

The answer and why: p. 467

Lesson 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.

The answer and why: p. 468

Lesson 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.

CASE STUDY · MODULE 1

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

An illustrative case, built from documented patterns.

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.

YOUR ANSWERS

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

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

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

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 58.

This page is blank so that how it played out stays out of view until you turn the page.

CASE STUDY · MODULE 1

How it played out

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.

The questions, discussed

1. 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.

2. 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.

3. 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. The answers, and the reason behind each, are in the key on p. 466. If two go wrong, re-read the module before the exam.

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

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.

Figure 2.1

Two cut-outs that look identical until the revision arrives

	On Monday	On Thursday
Selected the background and pressed Delete	**Identical** the pixels are gone	**Start again** back to the camera file, and every later edit with it
Selected the product and added a layer mask	**Identical** the pixels are hidden	**Eight seconds** a soft white brush along the base of the product

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,

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
- the delivered file

59 questions, the whole pool. The site draws 25 for a sitting, pass mark 70%. Work any 25 under your own conditions. Answers from page 518.

1 Module 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 Module 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 Module 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 Module 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 Module 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 Module 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

Module 1 · What an image is made of

The module starts on p. 11.

MODULE 1 TEST

Module 1 test, Q1, p. 60
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…
Answer C: Interpolation can only average samples that exist
Why: 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.

Module 1 test, Q2, p. 60
You want to build a mask from a single colour channel and need the one carrying the least noise. Why is green usually it?
Answer A: Sensors carry twice as many green photosites as red or blue
Why: 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.

Module 1 test, Q3, p. 60
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…
Answer B: Stopped later edits rounding onto the coarse grid
Why: 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.

Module 1 test, Q4, p. 61
A photograph of a starfield, downscaled to a quarter of its width, comes back noticeably darker than the original. What has gone wrong?
Answer D: Stored values were averaged instead of the light they represent
Why: 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.

Module 1 test, Q5, p. 61

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…

Answer A: The fringe comes from chroma subsampling, which happens before quantisation

Why: 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.

Module 1 test, Q6, p. 61

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?

Answer C: Two channels still hold real values when one has clipped

Why: 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.

THE LESSON CHECKS

Lesson 1, p. 16

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…

Answer 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.

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

A The upscaler was run at too…: Treats invention as an intensity dial rather than as what the model does at any setting.

B The file was exported as JPEG…: Blames the container for information that was never in the file to begin with.

D The image was left at 72…: Confuses the print-size metadata field with the actual pixel count.

Lesson 2, p. 20

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…

Answer 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.

Why: 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.

A The saturation slider damaged the file…: 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 Red and green happened to sit…: 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 The image was in the wrong…: 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.

Lesson 3, p. 23

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…

Answer C: The same gaps between the original tones, but no additional rounding damage from the Curves move and anything done after it.

Why: 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.

A Identical banding, because a JPEG can…: Over-corrects into fatalism: depth does not recover lost levels, but it does change how subsequent rounding accumulates, which is a real difference.

B Noticeably less banding, because 16-bit conversion…: Invents an interpolation step: conversion rescales existing values onto a wider axis and creates no intermediate tones between them.

D Worse banding, since spreading 256 values…: 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.

MAKE THINGS

Editing an Image

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

WHAT IT TEACHES	Raster editing as a working craft.
WHAT IS INSIDE	Eight modules and 78 lessons, 29 figures drawn from data, eight case studies, eight module tests, a 59-question examination, and every answer with the reason behind it.
LICENCE	
THE COURSE	Free to read, with the lessons, the films and the examination, at **addaly.com/learn/image-editing**

Addaly

First edition, September 2026 · build 62a26075

Scan to open the course