MAKE THINGS

Image Generation, in Practice

Reference images, inpainting, character LoRAs, upscaling, and the licence nobody read.

First edition, September 2026

Addaly

Series editor: Dr Mustafa Mun

Draft, open for academic review

MAKE THINGS

Image Generation, in Practice

Reference images, inpainting, character LoRAs, upscaling, and the licence nobody read.

Series editor: Dr Mustafa Mun

Addaly

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

Image Generation, in Practice

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). Image Generation, in Practice. Addaly. addaly.com/learn/ai-images.*

This book is the printed form of the course of the same name at addaly.com/learn/ai-images. 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.

7 modules · 69 lessons · 27 figures · 7 case studies · 55,057 words · about 9 hours

Brief contents

Contents 6

1 The kit, and what each piece of it costs you 9

2 From a request to a specification you can generate against 67

3 Getting to a frame worth keeping 117

4 Steering with pictures instead of adjectives 171

5 Repair, extend, assemble 219

6 The same thing, twelve times 271

7 Finishing and handing over 317

Examination 361

Answers 379

Contents

Module 1 · The kit, and what each piece of it costs you 9

1 The generator you pick is a licensing decision 11

2 A preset is somebody else's workflow, frozen 14

3 Three routes to a GPU, and the arithmetic for each 18

4 A model is four kinds of file, and mixing them up wastes an evening 23

5 Four front ends, and which one deserves your learning time 28

6 The first hour, and the four errors everybody gets 31

7 Steps, samplers and the seconds you should actually expect 36

8 Making a big model fit a small card 40

9 Every platform's filter is different, and that is a production risk 45

10 The setup that still works in six months 50

11 You are paying to make decisions you could make cheaply 54

Case study: Two hundred assets, and a licence nobody had read 58

Module 1 test 64

Module 2 · From a request to a specification you can generate against 67

12 Start at the placement, not the picture 69

13 The ratios the model was trained on, and what happens outside them 72

14 Designing for the crop you will not control 77

15 Six references, each with a job 81

16 Write down what must not appear 84

17 A hex value is not something a diffusion model can hit 88

18 One hero or twelve pieces is a different job 91

19 The frames where a phone camera wins 95

20 The one page that ends the revision argument 99

21 What you can sell, and what you must disclose 104

Case study: The coaching centre that wanted a campus it did not have 109

Module 2 test 114

Module 3 · Getting to a frame worth keeping 117

22 Build the prompt in slots, not as a sentence you keep rewriting 118
23 The camera vocabulary that works, and the words that do nothing 123
24 Name the light or accept the model's average 128
25 Describing a look without naming a living artist 132
26 Sixteen cheap frames beat four expensive ones 136
27 Fix the seed or learn nothing 140
28 Change one thing, or you have learned nothing 145
29 Recognising the twenty minutes that will never converge 149
30 Interrogating a picture for vocabulary 153
31 What "good enough to take forward" actually means 157
Case study: Forty minutes on seven boats 163
Module 3 test 168

Module 4 · Steering with pictures instead of adjectives 171
32 Stop describing it and show it the picture 172
33 The model is guessing your composition 176
34 Three passes down the denoise ladder 179
35 A biro sketch is the most reliable composition tool you own 184
36 Paint flat shapes, then let the model make them real 187
37 Get the perspective right by building it 191
38 Canny, lineart, depth, softedge, pose, normal 194
39 Different words for different parts of the frame 198
40 Transferring a look from an image without training anything 202
41 Two controls, without the plastic look 206
Case study: The hour of Blender nobody wanted to spend 211
Module 4 test 216

Module 5 · Repair, extend, assemble 219
42 The finished image is never one generation 220
43 Fix it in the order that stops you doing it twice 224
44 The three repairs everybody needs, with settings 229
45 Type is set, not generated 233
46 Extending a frame when the crop changes 238
47 Four ways to remove an object, and when each is right 242
48 The real product in a generated world 245

49 Making two sources agree 250
50 The fault the client sees on a big screen 254
51 Getting to poster size without a second horizon 257
Case study: A 4:5 frame, a 16:9 banner, and two hours 262
Module 5 test 268

Module 6 · The same thing, twelve times 271
52 The same face twice is the hard part 272
53 Build the reference before you build the images 275
54 Training a character LoRA on hardware you can borrow 279
55 Four dataset faults, and the LoRA each one ruins 283
56 The decision, with the numbers 286
57 A product is harder than a face 290
58 Five constants that do more than the face 294
59 A file scheme that survives the revision round 299
60 A worked run, with the budget and the failure 304
Case study: Twelve frames, one weaver, and a photograph nobody had asked permission for 309
Module 6 test 314

Module 7 · Finishing and handing over 317
61 Upscaling either restores detail or invents it 318
62 Nothing goes to a client straight from the model 321
63 300 dpi is one number applied to everything 325
64 The blue on your screen is not going on the press 329
65 Making words readable on a photograph 333
66 Twelve things to check before you send 336
67 Exporting without wrecking it at the last step 340
68 Describing the image, including what it is 344
69 When "just change one thing" is a rebuild 348
Case study: Seventy megapixels nobody could see 352
Module 7 test 358

Examination 361

Answers 379

1

MODULE 1

The kit, and what each piece of it costs you

Assemble a working image-generation setup from the hardware you actually own — hosted service, local install or a free cloud GPU — and state for your chosen route the VRAM it needs, the seconds and rupees per image it costs, the licence it carries, and the exact files on disk that make up the model.

1 The generator you pick is a licensing decision 11
2 A preset is somebody else's workflow, frozen 14
3 Three routes to a GPU, and the arithmetic for each 18
4 A model is four kinds of file, and mixing them up wastes an evening 23
5 Four front ends, and which one deserves your learning time 28
6 The first hour, and the four errors everybody gets 31
7 Steps, samplers and the seconds you should actually expect 36
8 Making a big model fit a small card 40
9 Every platform's filter is different, and that is a production risk 45
10 The setup that still works in six months 50
11 You are paying to make decisions you could make cheaply 54
Case study: Two hundred assets, and a licence nobody had read 58

Module 1 test 64

Lesson 1 · 9 min

The generator you pick is a licensing decision

THE ONE THING TO KEEP

The licence on a model's weights governs whether you may use it commercially, and it is a completely separate question from who owns the picture it made.

Everyone compares the pictures. Almost nobody compares the terms

Pick any two image generators and the sample galleries will look roughly as good as each other. That is not where they differ any more. They differ on three things that decide whether you can actually finish a job: what one image costs you, how much control you get when the first result is wrong, and what you are permitted to do with the file afterwards.

The third one is the one that ends careers-in-miniature — a designer who delivers 200 assets built on weights they were not licensed to use commercially.

If you have not read how these models actually work, read that first. This course assumes it and never repeats it.

The hosted tools, and what each is actually for

Midjourney still produces the most immediately attractive frame with the least effort. The cost of that is an opinion baked into every image — a house look you have to fight. It is subscription-only, with no free tier, and on the entry plan your generations are visible in the public gallery; private generation sits on the higher tiers. If you work under NDA, check that before your first render, not after.

Adobe Firefly is the conservative choice. It is trained on Adobe Stock and licensed or public-domain material, and Adobe offers indemnity against training-data claims on its business plans. It is less exciting and more

defensible. It has a free monthly credit allowance, and Generative Fill inside Photoshop is the same engine.

Google's Gemini and Imagen line is strong at following a long instruction and at editing an image you supply. Google stamps its generated images with SynthID, an invisible watermark. There is meaningful free use through the Gemini app and AI Studio.

Ideogram exists because text inside images was broken for years. It renders legible words far better than the general-purpose models. It is still not typesetting — more on that in the finishing lesson.

Recraft aims at design output rather than pictures: it will give you actual vector SVG, which matters if the thing you are making has to scale to a billboard. See vector and print for why that distinction is not cosmetic.

Higgsfield is a different species — a recipe-driven platform rather than a raw model. The next lesson is about that whole category.

Running the weights yourself

The open-weights line runs Stable Diffusion → SDXL → SD 3.x, alongside FLUX from Black Forest Labs and Qwen-Image from Alibaba. Running them yourself costs nothing per image, gives you every control that exists, and works offline.

The requirement is a GPU. SDXL is comfortable on 8 GB of VRAM. FLUX's larger variants want much more, though quantised versions run on 8-12 GB slowly. If you do not own a card: Google Colab's free tier, Kaggle's free weekly GPU hours, and Hugging Face Spaces all run these models at no cost. Running models yourself and Hugging Face cover the mechanics.

Free front ends, all genuinely free: **ComfyUI** (a node graph, ugly, total control), **Fooocus** (Midjourney-like defaults, almost no settings), **InvokeAI**, and **Krita with the AI Diffusion plugin**, which is the best free inpainting experience there is.

The licence trap, stated plainly

A model's weights carry a licence, and it is a separate thing from what you may do with the pictures.

- **FLUX.1 [schnell]** is Apache 2.0. Do what you like.
- **FLUX.1 [dev]** is a non-commercial licence. Thousands of people have downloaded it, run it locally, and billed clients for the output. That is a breach of the model licence, regardless of any question about who owns the image.
- **Qwen-Image** was released Apache 2.0.
- **Stability's recent releases** use a community licence with a revenue threshold: free below it, paid above.
- **Hosted tools** replace all this with their terms of service, and free tiers frequently restrict commercial use even when paid tiers do not.

Those were the positions in 2025 and they move. Check the model card on the day you start the job. Lesson ten deals with output rights, which is a different question again.

Choosing, in about thirty seconds

- A single striking image, no budget for fiddling — Midjourney or the Gemini app.
- Corporate work where somebody will ask about training data — Firefly, or Getty's and Shutterstock's own generators.
- Words inside the image — Ideogram, then set real type over it anyway.
- Hundreds of near-identical variants — a preset platform.
- Precise control, a repeating character, or no money at all — open weights, locally or on free GPU hours.

Today: open the model card or terms page of whatever you are currently using and find the sentence about commercial use. Most people have never read it.

BEFORE YOU MOVE ON

A freelancer downloads FLUX.1 [dev], runs it on their own machine with no internet connection, and invoices a client for the resulting images. What is the actual problem?

A. Nothing is wrong. A licence attached to model weights controls redistributing the file, not what you make with it once it is on your disk.

B. The images are unlikely to attract copyright, so there was nothing there to sell in the first place.

C. The [dev] weights are released under a non-commercial licence, so billing a client breaches the model's terms whatever the copyright position on the images turns out to be.

D. Generating locally strips the Content Credentials the client needed for disclosure, which is what makes the delivery non-compliant.

The answer and why: p. 381

Lesson 2 · 8 min

A preset is somebody else's workflow, frozen

THE ONE THING TO KEEP

Recipe-driven tools trade control for speed, and the bill arrives at the first revision, when the thing the client wants changed is the thing the preset already decided.

Why the packaged tools feel like magic

Open a raw model and you face a blank prompt box, a sampler dropdown, a step count, a guidance scale, an aspect ratio and a seed. Most of those decisions have one sensible answer for the thing you are making, and you do not know what it is yet.

Open a platform like **Higgsfield** and you face a wall of named recipes instead — a product-shot look, a marketing card layout, a cinematic portrait, a set of camera moves borrowed from its video side. You click one, you drop in a photo or a line of text, and something usable comes out in under a minute.

That is not a cheap trick. A preset is a real workflow that somebody built and tested: a chosen model, a prompt scaffold you never see, probably a style adapter or a LoRA, a fixed aspect ratio, a lighting description, sometimes an upscale pass on the end. The speed is genuine. So is the price you pay for it.

What Higgsfield-shaped tools are good at

Three jobs, and they are good at all three.

Volume with a consistent look. Forty product cards a week that must belong to the same family. A raw model gives you forty siblings that do not match. A preset gives you forty that do, because the same hidden scaffold is under each one.

Character identity as a saved object. The interesting feature in this category is identity training — Higgsfield calls its version Soul ID. You supply a set of photographs of a person once, the platform trains an identity, and then that person can appear across many generations without you rebuilding the setup each time. Under the surface this is the same idea as a LoRA, which lesson six covers properly; the platform difference is that it is three clicks instead of a training script.

Camera language as a control. Because tools like this grew out of video, they expose motion and lens vocabulary as buttons — crash zoom, dolly, low angle, bullet time — rather than expecting you to write it into a prompt and hope. If you do not have a cinematography vocabulary yet, this teaches you one by letting you see each term applied to your own frame.

What you give up, and when you notice

You notice at the revision.

The client likes the bottle shot but wants the light coming from behind the glass instead of the front. That decision was made inside the preset. It is not in your prompt, so no amount of rewriting the prompt reaches it. The preset's opinion about lighting is applied after your words and overrides them. You cannot get to a point halfway between two presets, either — they are discrete choices, not sliders.

This is the general rule and it is worth carrying beyond this one tool: **anything that packages a workflow buys you speed by removing exactly the controls you will need when the first answer is wrong.** It is a good trade for the ninety per cent of work that is routine, and a bad trade for the ten per cent that is the reason a client hired a person.

The credit arithmetic nobody checks

Platforms in this category run on credits, and within one platform the per-image credit cost across the available models can differ by nearly an order of magnitude.

A concrete measurement, from a real project in mid-2026: generating small avatar portraits on one such platform, a lightweight model cost 0.15 credits per image while two premium models cost 1.00 and 1.25. Seven to eight times the price. At the size the images were actually used — a 48-pixel avatar — no one could tell them apart.

The lesson generalises. Before you commit a batch, generate the same frame on the cheapest and the dearest model the platform offers, then look at both **at the size they will be seen**. Half the time the expensive one wins clearly and is worth it. The other half you have just found a way to make your allowance last eight times longer.

The free path

There is no free tier that gives you a preset platform. What there is, and what does the same job: **Fooocus**, which is a free local front end built on exactly this philosophy — good defaults, almost no visible settings, style presets in a dropdown — and runs on free Colab or Kaggle GPU hours if you have no card.

Canva's free tier packages generation into templates in a comparable way for social and print layouts, and works on a phone. Neither will give you trained character identity as a one-click object; you build that yourself in lesson six.

Use one honestly

A preset tool is not a lesser tool. It is a tool with a shape. Use it when the work is repetitive and the look is agreed. Reach for raw model controls when a specific person has a specific note.

Today: pick any preset you like the output of, then try to produce the *same* image with one deliberate change — a different light direction, a different lens. How far you get tells you exactly how much control that platform kept for itself.

BEFORE YOU MOVE ON

A team produces 40 product cards a week on a preset platform with no complaints. Then a client asks for one card where the light comes from behind the bottle rather than the front, and nobody on the team can produce it. What does this actually reveal?

A. The underlying model is too small to follow lighting instructions, so the team needs a platform running a larger model.

B. The preset has already fixed the lighting decision and applies it regardless of the prompt, so the fix is a tool that exposes lighting directly, or a manual relight in a raster editor.

C. The prompt needs to name the backlighting explicitly and in more detail than it currently does.

D. The preset needs retraining on backlit examples before it can produce them.

The answer and why: p. 382

Lesson 3 · 9 min

Three routes to a GPU, and the arithmetic for each

THE ONE THING TO KEEP

The choice between a hosted service, your own card and a free cloud notebook is decided by how many images you make per month and how much control you need, and the crossover is usually somewhere around a few thousand images.

The same picture, three different bills

There are three ways to get a diffusion model to run, and people pick between them on feel when the decision is arithmetic.

Route one: a hosted service. You type into a website or call an API and pay per image. Typical published prices at the time of writing sit between roughly $0.01 and $0.06 an image depending on model and resolution, with the cheap end being small distilled models and the expensive end being the current flagship. No installation, no driver, no disk space. Your phone is enough.

Route two: your own graphics card. You install the software once and then generate for the cost of electricity — about 250 watts for a few seconds an image, which is a rounding error. The entry barrier is the card. Useful floors:

- **4 GB VRAM** — Stable Diffusion 1.5 at 512×512. Slow, dated, but it works.
- **8 GB** — SDXL at 1024×1024, comfortably. SD 1.5 at speed. FLUX only with quantisation and offloading, at maybe 40 seconds an image.
- **12–16 GB** — FLUX dev in fp8, SDXL with several adapters loaded at once, LoRA training.
- **24 GB** — everything, including training at sensible speeds.

An Apple Silicon Mac works through Metal rather than CUDA. It is slower than an equivalent Nvidia card — often three to five times — but unified memory means a 16 GB MacBook can load models that would not fit an 8 GB desktop card. It is a real option, not a consolation prize.

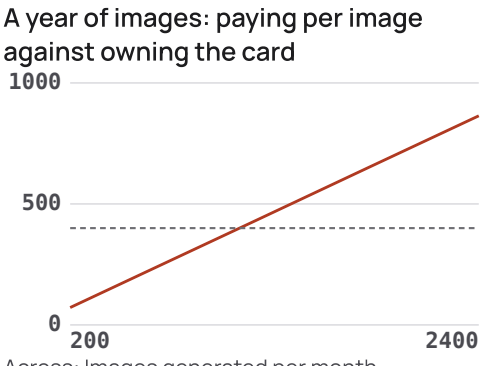
Route three: somebody else's GPU, free. Google Colab's free tier gives you a T4 with 16 GB, with sessions that disconnect after a few hours and no guarantee of availability. Kaggle Notebooks give roughly 30 GPU-hours a week on a T4 pair or a P100, which is more generous and less advertised. Hugging Face Spaces run other people's apps on shared hardware, and the ZeroGPU tier gives short bursts of an A100 for free. All three are genuinely free and all three have the same catch: nothing persists. You reinstall every session, you re-download the model every session, and a 23 GB checkpoint download eats twenty minutes of your allowance before you have made a single picture.

Where the crossover sits

Do the sum before you buy anything. Suppose you make 200 images a month at $0.03. That is $6 a month, $72 a year. A used card with 12 GB of VRAM costs several hundred dollars. At that volume, hosting is cheaper for years.

Now suppose you are exploring: 40 variations to find one frame, 15 frames a week. That is 2,400 images a month, $72 a month, and the card pays for itself inside a year while also letting you run things no service offers.

Figure 1.1



Across: Images generated per month
Up: Cost over a year, in dollars
—— Hosted, three cents an image
– – A used 12 GB card, bought once

The lines cross a little above a thousand images a month. Below that, hosting is cheaper for years. Above it, the card pays for itself inside a year — and most people who buy one do it for the control rather than the arithmetic.

The honest version of this is that most people who buy a card do it for the second reason and then discover the third: **control**. A hosted service decides which model versions exist, what the safety filter blocks, and whether your workflow still runs next Tuesday. A card you own does none of that to you.

The constraint nobody mentions

VRAM is not the only limit; it is just the one that produces a clear error message. The quieter limit is **system RAM during model loading**. Loading a 23 GB checkpoint usually means reading it into system memory first, and a machine with 8 GB of RAM will swap to disk and take four minutes to do what a 32 GB machine does in twenty seconds. People blame the GPU for this constantly. If your first generation after a model switch is glacial and later ones are fine, it is loading, not inference.

The second quiet limit is **disk**. A working local setup with three checkpoints, a VAE, a dozen LoRAs and a control model collection passes 100 GB without effort. On a 256 GB laptop that becomes a weekly problem.

What to do first

Do not buy anything. Spend one week on a free Kaggle notebook or a hosted service with a small credit, and count two things: how many images you generate, and how many times you are blocked by something the platform will not let you do. Those two numbers pick your route, and they are cheaper to collect than to guess.

The hybrid most working people end up with

Almost nobody stays on one route. The arrangement that settles out after a few months looks like this, and it is worth arriving at deliberately rather than by accident:

- **Exploration on a fast cheap model**, hosted or local, where you are making forty frames to find one composition and the quality of each is irrelevant.
- **The final render wherever the quality is**, which for a flagship look means a hosted API, and for anything the filter dislikes means your own weights.
- **Repair and compositing in a raster editor**, which needs no GPU at all. GIMP, Krita and Photopea run on the laptop you already have, and a surprising share of the finished work happens there rather than in any generator.

Splitting the job this way also protects you from the thing that breaks single-route setups: a service changing its pricing, its model or its policy. If exploration and final render are already two different tools, replacing one of them is an afternoon rather than a crisis.

One caution about the free cloud notebooks, because it catches people who are being careful about money. Colab and Kaggle both restrict some categories of use in their terms, and both will disconnect a session that looks unattended or permanently loaded. Using them interactively within the published limits is fine and intended. Wiring one up as a standing server for a client is not, and losing the account takes the project with it.

BEFORE YOU MOVE ON

Somebody generates about 150 images a month and keeps hitting a hosted service's refusal filter on ordinary historical war imagery for a school project. Which factor should decide their next step?

A. Control, not cost: at this volume hosting is a few dollars a month, so the real question is whose policy decides what they may make.

B. Cost, because 150 images a month is a steady enough workload that owning the hardware outright will repay the purchase price of a mid-range graphics card within a reasonable period.

C. They should move to a free Colab or Kaggle notebook, which removes the running cost and the platform's content filter at the same time and therefore addresses both halves of the problem at once.

D. They should rewrite the prompts, since a refusal is a reliable signal that the wording was ambiguous and a clearer, more specific description will pass the check.

The answer and why: p. 382

Lesson 4 · 8 min

A model is four kinds of file, and mixing them up wastes an evening

THE ONE THING TO KEEP

A checkpoint, a VAE, a LoRA and an embedding are different file types with different sizes and different folders, and almost every "it loaded but the image is grey mush" problem is one of them in the wrong place.

The inventory on disk

People say "I downloaded the model" as if a model were one thing. Locally it is a small family of files, and knowing which is which saves you the evening where everything loads and every image comes out as grey porridge.

The checkpoint is the model proper: the denoising network plus, usually, the text encoder and VAE bundled in. It is the big file.

- SD 1.5 checkpoint: 2–4 GB
- SDXL checkpoint: about 6.6 GB
- FLUX dev in bf16: about 23 GB; the fp8 version about 11 GB; a GGUF Q4 version about 6.5 GB

The VAE is the decoder that turns the small latent image into pixels. Around 160–330 MB. Most checkpoints have one baked in. You only need a separate one when a checkpoint shipped with a broken or "pruned" VAE, and the symptom is unmistakable: correct composition, correct colours in the thumbnail preview, and a final image that is washed out, purple-tinged or covered in noise. Swap the VAE and it is fixed. This is the single most common "the model is broken" that is not.

A LoRA is a small set of weight adjustments layered onto a checkpoint — a character, a style, a concept. 20–250 MB. A LoRA is built for a specific base: an SD 1.5 LoRA on an SDXL checkpoint either errors or does nothing useful.

The mismatch is the second most common wasted evening.

A textual inversion or embedding is smaller still, often 4–80 KB. It teaches the text encoder one new word rather than changing the model. Mostly used now for negative-prompt bundles.

There are also **ControlNet models** (1.4 GB each for SD 1.5, up to 2.5 GB for SDXL) and **IP-Adapter** files. These are conditioning modules, again base-specific.

Figure 1.2

The five kinds of file, and the evening each one costs you

Checkpoint — the model proper
2–23 GB. In the wrong folder, nothing loads at all.

VAE — the decoder to pixels
160–330 MB. A broken one gives grey mush or a purple cast.

LoRA — a difference from one checkpoint
20–250 MB. Wrong base: it errors, or quietly does nothing.

Embedding — one new word for the text encoder
4–80 KB. Mostly negative-prompt bundles now.

ControlNet and IP-Adapter — conditioning modules
1.4–2.5 GB. Base-specific, exactly like a LoRA.

Every one of them is base-specific, and almost every "the model is broken" report is one of them in the wrong folder or against the wrong base.

safetensors, not ckpt

You will see two extensions. A `.ckpt` file is a Python pickle, which means loading it executes code contained in the file. That is not a theoretical risk; malicious checkpoints have been distributed. A `.safetensors` file is a plain tensor container that cannot execute anything.

The rule is simple: **download safetensors**. If only a `.ckpt` exists and you need it, convert it in an isolated environment, or find another copy. No workflow is worth handing a stranger a shell on your machine.

Where they go

Folder layouts differ between interfaces, but the shape is the same everywhere, and this is roughly what a Forge or A1111 install looks like:

```
models/
  Stable-diffusion/    <- checkpoints (.safetensors)
  VAE/                 <- standalone VAE files
  Lora/                <- LoRAs
  embeddings/          <- textual inversions
  ControlNet/          <- control models
```

ComfyUI uses `models/checkpoints`, `models/loras` and so on, and has an `extra_model_paths.yaml` that lets a second interface read the same folders. Set that up on day one if you plan to use more than one front end. Storing a 6.6 GB checkpoint twice is how a disk fills.

The limitation worth stating

A LoRA is trained against a specific checkpoint, but it is usually *used* on a different one — someone trains a character on base SDXL and you apply it to a heavily fine-tuned photographic checkpoint. It often works, which is why people do it. It also frequently degrades: colours shift, contrast rises, the face drifts toward the training checkpoint's house style.

The mechanism is that a LoRA encodes a *difference* from a particular set of weights. Applied to weights that have already moved, the difference lands somewhere else. This is why the same LoRA at strength 0.8 looks right on one checkpoint and plasticky on another, and why the answer is to lower the strength to 0.5–0.6 rather than to conclude the LoRA is bad.

Merges, and why your favourite checkpoint has no history

A large share of the checkpoints people actually use are **merges**: two or more models averaged together, sometimes with a LoRA baked in, often by somebody who did it by feel in an afternoon. Merging is legitimate and some merges are excellent. It has three consequences you should hold in mind.

The first is that a merge inherits every licence in its ancestry, and the upload page frequently does not say what those were. If you are working commercially, a merge with an unclear lineage is a risk you are carrying without knowing its size.

The second is that merges drift toward a house look. Average several photographic checkpoints and you get the same glossy, slightly over-lit, slightly symmetrical face that everybody's images have. If your output looks generically "AI", the checkpoint is often more responsible than the prompt.

The third is practical: merges frequently ship a broken or non-standard VAE, which is why the grey-porridge and purple-cast problems cluster on community downloads rather than on official base models.

What to actually download first

Start with one official base model and one well-documented fine-tune, and nothing else. A first shelf that works:

- One base checkpoint appropriate to your VRAM.
- Its matching official VAE, kept separately so you can swap it in when a merge misbehaves.
- One upscaler model, a few tens of megabytes.

Add a LoRA only when you have a specific job for it. The usual failure of a new setup is fifteen checkpoints, ninety LoRAs, a full disk and no finished images.

BEFORE YOU MOVE ON

An SDXL image has correct composition and correct framing but comes out washed out with a purple cast on every generation, regardless of prompt or seed. What is the most likely cause?

A. A LoRA trained against a different base model is being applied on top of this checkpoint and is corrupting the output as it is blended in.

B. The VAE is broken or missing, so a correctly denoised latent is being decoded wrongly.

C. The guidance scale has been set too high for this checkpoint, which oversaturates the image and shifts the colour balance as it clips.

D. The checkpoint was downloaded as a .ckpt rather than a .safetensors, and that container stores its weights at a lower numeric precision.

The answer and why: p. 383

Lesson 5 · 8 min

Four front ends, and which one deserves your learning time

THE ONE THING TO KEEP

The interface you learn decides which problems are easy, and the right choice is the simplest one that can express the workflow you actually need — which for finished client work usually means a node graph eventually.

They all call the same model

Every local interface runs the same weights through roughly the same code. The differences are entirely about what they make easy and what they make impossible, and choosing wrongly costs weeks.

Fooocus hides almost everything. You type a prompt, pick a rough style, and it applies a tuned set of defaults that are genuinely good. Installation is close to one step. It is the fastest route from nothing to a decent SDXL image on a Windows machine, and it is an excellent first week. It is also a ceiling: when you need a second ControlNet or a custom step order, it cannot express that.

AUTOMATIC1111 and its faster fork, Forge, give you a form. Prompt, negative prompt, sampler, steps, seed, and tabs for img2img, inpainting and extensions. The mental model is a settings panel, which is why most tutorials are written for it. Forge is the one to install today: same interface, noticeably better memory handling, so SDXL fits on 6 GB cards that choke under A1111.

InvokeAI sits between the two, with a real canvas. If your work is mostly inpainting and outpainting on a single image — retouching, extending, iterating on one picture — its unified canvas is the best tool of the four, because the thing you actually do all day is "mask this bit and regenerate", and Invoke makes that one gesture rather than five.

ComfyUI is a node graph. You wire a checkpoint loader into a sampler into a decoder into a save node. Nothing is implicit. The learning curve is real — most people bounce off it once — and the payoff is that a workflow becomes a file you can save, share, version and re-run in eight months and get the same result.

The honest recommendation

If you are learning: start with Forge, because the tutorials match and the feedback loop is short.

If you are delivering paid work: you will end up in ComfyUI, and starting there costs you a weekend rather than a month. The reason is not power for its own sake. It is that client work means running the same eleven-step process on image after image, and a form-based interface stores that process in your head and your notes, where it decays. A saved graph does not decay.

If your work is one image at a time, heavily hand-repaired: InvokeAI, and do not apologise for it.

The trap in shared workflows

ComfyUI's great convenience is that a workflow is embedded in the PNG a stranger posted, so you can drag their image onto your canvas and get their exact graph. The trap is **custom nodes**. That graph may reference twelve community node packs, three of which are abandoned and one of which conflicts with something you already have. You spend an hour in the manager resolving red boxes, and then the output still differs because their checkpoint is a merge nobody has re-uploaded.

The habit that prevents this: when a downloaded workflow works, save your own cleaned-up copy with the minimum nodes, and note the model hashes alongside it. Treat the stranger's graph as a reference, not a dependency.

One more thing worth knowing before you choose. All four of these are free and open source, and all four update aggressively. An interface that updates weekly will, at some point, change a default you were relying on — a scheduler

renamed, a node's output reordered — and yesterday's look will not reproduce. This is a real cost of the free tooling and the answer is in the last lesson of this block: pin your versions and write them down.

Where the paid tools sit in this

Job adverts name Photoshop, and Photoshop's generative fill is a competent, well-integrated inpainting tool with a commercially indemnified model behind it. If a workplace already pays for it, use it; there is nothing to prove by refusing.

What matters is that none of the four interfaces above is a poor relation. They run the same class of model, and on several axes — conditioning types, workflow reuse, model choice, cost per image — the free stack does more than the subscription one. A freelancer with ComfyUI, GIMP and Inkscape on a six-year-old laptop can deliver work a studio on a full Adobe licence cannot, because the studio's tool only offers the models its vendor chose to license.

The honest split is this. Paid tools are better at *integration*: one application, one file, a predictable undo history, and somebody to complain to. Free tools are better at *reach*: any model, any conditioning, any step order, no per-image cost.

Do not learn all four

The most common mistake in the first month is installing every interface to compare them. You end up with four half-understood tools, 180 GB of duplicated checkpoints, and no finished work.

Pick one on the rule above, and give it two weeks without looking sideways. The skills that actually transfer — reading denoise strength, judging a mask edge, knowing when a frame is unrecoverable — are not interface skills at all, and every hour spent on the interface question is an hour not spent acquiring them.

BEFORE YOU MOVE ON

A studio produces the same eleven-step retouch-and-composite process on roughly forty product images a month, handed between two people. Which interface choice best fits, and why?

A. Fooocus, because its curated defaults produce a consistent house look without either operator having to configure samplers or step counts by hand.

B. ComfyUI, because the eleven steps become one saved graph both people run identically.

C. AUTOMATIC1111 or Forge, because a settings form is faster to operate than a node graph once the eleven steps have been committed to memory by both operators.

D. InvokeAI, because its unified canvas is the strongest inpainting and outpainting surface of the four and this is fundamentally retouching work.

The answer and why: p. 383

Lesson 6 · 8 min

The first hour, and the four errors everybody gets

THE ONE THING TO KEEP

Almost every failed local install is one of four specific errors, each with a one-line cause, and knowing them turns a lost weekend into twenty minutes.

Install once, properly

A local setup is a Python environment, a PyTorch build matched to your hardware, an interface, and at least one checkpoint. Most guides cover the happy path. This lesson covers the four places it stops.

Use a virtual environment. Always. Not because it is tidy, but because two interfaces will want incompatible versions of the same library, and without isolation the second install silently breaks the first.

```
python -m venv venv
source venv/bin/activate      # Windows: venv\Scripts\activate
python --version              # 3.10 or 3.11; 3.13 still breaks
things
```

The Python version matters more than people expect. Much of this ecosystem pins to 3.10, and installing on the newest release is a reliable way to meet a compiled dependency that has no wheel for your version and tries to build from source.

Error one: "Torch not compiled with CUDA enabled"

You installed PyTorch with a plain `pip install torch`, which on many systems gives you the CPU-only build. Generation then either refuses or runs at roughly one image every four minutes.

The fix is to install the build that matches your hardware, from PyTorch's own index:

```
# Nvidia
pip install torch torchvision --index-url
https://download.pytorch.org/whl/cu121
# AMD on Linux
pip install torch torchvision --index-url
https://download.pytorch.org/whl/rocm6.0
# Apple Silicon: the default build already has Metal support
```

Verify before you go further. `python -c "import torch; print(torch.cuda.is_available())"` printing `False` on an Nvidia machine means stop and fix this now, not after installing four extensions.

Error two: "CUDA out of memory"

The message names a number: it tried to allocate, say, 2.14 GiB and had 1.87 GiB free. That number is the useful part. In order of what to try:

1. Lower the resolution. Memory scales roughly with pixel count, so 1024×1024 to 768×768 is a 44% cut.
2. Reduce batch size to 1. A batch of four costs close to four times the activation memory.
3. Enable medium or low VRAM mode (`--medvram`, `--lowvram` in Forge; `--lowvram` in ComfyUI).
4. Turn on tiled VAE. Decoding is often the peak, not sampling — which is why a generation gets to 100% and *then* crashes.

That last point is the one people miss. If the progress bar completes and the failure comes after, it is the decoder, and tiled VAE fixes it without touching anything else.

Error three: the mismatched extension

An extension installs a different version of a library the interface already uses, and the whole thing stops starting. The symptom is a stack trace naming a package you never installed deliberately.

The cure is prevention: install extensions one at a time, and start the interface after each one. When something breaks you know exactly what did it, and you can delete that one folder. Installing six and then debugging is a genuinely miserable afternoon.

Error four: it works, and looks nothing like the tutorial

Your setup is fine. You have a different checkpoint, or a different sampler default, or the tutorial used a hidden style preset. Diffusion output is extremely sensitive to configuration, and "wrong" here almost never means broken.

Before concluding anything, reproduce a known image: take a picture with embedded metadata from a model's own page, copy every parameter including sampler, steps, CFG, seed, size and model hash, and generate. If you get their image, your install is correct and your differences are parameters. If you do not, compare hashes — you are probably on a different checkpoint with the same name, which happens constantly because everyone re-uploads everything.

The one that is not your fault

On Windows with an Nvidia card, the driver will, by default, spill VRAM into system RAM instead of raising an out-of-memory error. Generation does not crash; it becomes twenty times slower, with no message. If your speed collapses after loading a second model and nothing errors, this is it. Turn off "CUDA system memory fallback" in the Nvidia control panel and you will get a clean error instead of a mystery.

The half hour that pays for itself

Once it runs, do three things before you start real work.

Generate a known image and keep it. Pick one prompt, one seed, one checkpoint, and save the result as `reference.png` in the install folder. After every update, regenerate it. If it differs, something changed, and you have found out in ten seconds rather than in the middle of a client revision.

Write down where the models live. One text file with the folder paths and what goes in each. You will be adding a second interface within a month, and pointing it at the same folders instead of downloading everything again saves tens of gigabytes.

Turn on metadata saving and PNG output. Most interfaces do this by default, but check. A JPEG delivered to a client is disposable; a PNG with the full parameter block in it is the only record of how the image was made, and the next block's whole working method depends on having it.

Where to ask when it breaks

An install failure is almost never unique to you. Paste the exact final line of the error into a search, not your description of the symptom. "It doesn't work" finds nothing; `AssertionError: Torch not compiled with CUDA enabled` finds the answer as the first result, because several thousand people have already hit it. The project issue trackers on GitHub are searchable and usually contain a closed issue with the fix in the second comment.

BEFORE YOU MOVE ON

A local SDXL generation reaches 100% on the progress bar and then fails with a CUDA out-of-memory error. What does the timing tell you?

A. Nothing useful, because an out-of-memory failure has one cause and one remedy, which is to generate the image at a smaller resolution and try again.

B. The batch size is too high, since a batch allocates memory for all of its images together at the point where they are written out.

C. The peak is the VAE decode that runs after sampling, so tiled decoding is the targeted fix.

D. The checkpoint itself is too large for this card, so it has to be quantised to a smaller format before any other change will make a difference.

The answer and why: p. 384

Lesson 7 · 8 min

Steps, samplers and the seconds you should actually expect

THE ONE THING TO KEEP

Doubling the step count past roughly 30 buys almost nothing on a standard model, while a distilled model does a usable image in 4 to 8 steps, so measure iterations per second and spend your time where the curve is still steep.

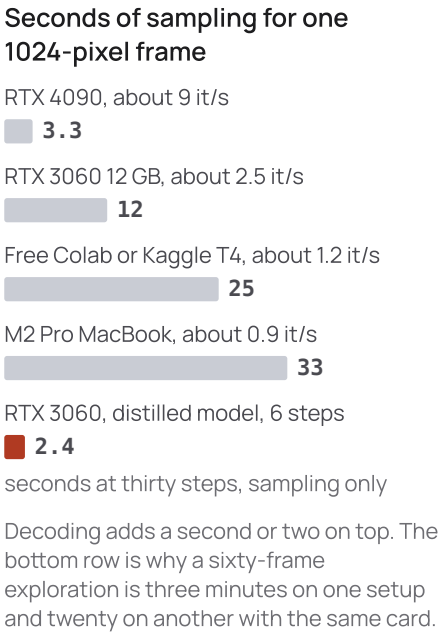
Know your it/s

Every interface prints a speed figure during generation, usually iterations per second, and it is the only number that lets you predict anything. Ballpark figures at 1024×1024 on SDXL:

- RTX 4090: roughly 8–10 it/s
- RTX 3060 12 GB: roughly 2–3 it/s
- T4 (free Colab, free Kaggle): roughly 1–1.5 it/s
- M2 Pro MacBook: roughly 0.7–1.2 it/s

At 30 steps, 2 it/s means 15 seconds of sampling plus a second or two of decoding. That is your planning unit. If you intend to explore 60 variations, you are committing to about twenty minutes, and that fact should change how you explore.

Figure 1.3



Decoding adds a second or two on top. The bottom row is why a sixty-frame exploration is three minutes on one setup and twenty on another with the same card.

The step curve flattens, and where it flattens

Steps are how many denoising passes the sampler makes. More steps means each one moves less, and the result converges. The useful shape of the curve, for a standard non-distilled model:

- 1–10 steps: visibly unfinished, soft, sometimes structurally wrong
- 15–25 steps: the image is essentially decided
- 25–35 steps: small refinements to fine texture
- 35–80 steps: differences you need to A/B at 200% zoom to see
- past 80: nothing, and occasionally worse, as some samplers drift

So 60 steps costs twice 30 and buys almost nothing. Where people go wrong is inferring "more steps equals more detail". It does not add detail; it finishes the detail already implied by the seed and prompt.

The important exception is at very low step counts with certain samplers. Ancestral samplers (the ones with an `a` — Euler a, DPM++ 2S a) inject fresh noise at every step, so they never fully converge: the image at 20 steps and at 40 steps are genuinely different pictures, not the same picture refined. That is

why a seed that looked perfect at 25 steps changes when you push to 40 for the final render. If you want "same image, more finish", use a non-ancestral sampler such as DPM++ 2M with the Karras schedule, and the seed stays stable as steps rise.

Distilled models change the arithmetic

Turbo, Lightning, LCM and Hyper variants are trained to reach a usable image in very few steps — typically 4 to 8, sometimes 1. They want low guidance too, often 1.0 to 2.0 rather than the usual 5 to 8. Get that wrong and the output looks burnt and over-contrasted, which is the single most common reason people conclude a turbo model is bad.

A distilled SDXL at 6 steps on a 3060 takes about three seconds. That turns a 60-image exploration from twenty minutes into three, which is not a convenience — it is a different way of working, and the next block builds a whole method on it.

The cost is real and worth naming: distilled models have less variety. The distillation process teaches the model to jump straight to a high-probability output, so different seeds land closer together, and unusual prompts get pulled toward the ordinary interpretation. Use them to choose composition, then render the winner on the full model.

Where the time actually goes

For a single image people assume sampling dominates. Often it does not.

- Model load from disk: 5–60 seconds, once per model
- Text encoding: well under a second
- Sampling: steps ÷ it/s
- VAE decode: 0.5–3 seconds, more at high resolution
- Saving with metadata: negligible

Switching checkpoints between every generation can easily spend more time loading than generating. If you are comparing two checkpoints, do all twelve images on the first and then all twelve on the second, rather than alternating.

People alternate because it feels more rigorous. It is the same comparison, four times slower.

A default that is fine to stop thinking about

There are roughly thirty samplers in a modern interface and the differences between the good ones are small. A defensible default that you can stop revisiting:

- **DPM++ 2M with the Karras schedule, 25–30 steps, CFG 5–7** for SDXL and most fine-tunes.
- **Euler, 20 steps, guidance 3.5** for FLUX-family models, which behave differently and dislike high CFG.
- **The model card's own numbers** for anything distilled, because those are not suggestions.

Change one of these when you have a reason — you want the ancestral variation, or a model card specifies otherwise — not as a way of hoping for a better picture. Sampler roulette is where beginners spend their first fortnight, and the gain over a sensible default is smaller than the gain from one better sentence in the prompt.

The honest limit of the speed figures

Published it/s numbers are close to useless for comparing your machine with somebody else's, because they depend on resolution, precision, attention implementation, batch size, and whether the card is thermally throttling in a laptop chassis. A laptop 4070 can run at half the speed of a desktop 4070 under sustained load, and neither figure is wrong.

Measure your own, once, at the resolution you actually use, and write it down. That single number tells you what a sixty-image exploration costs in minutes, and that is the only thing it was ever needed for.

BEFORE YOU MOVE ON

A frame looks right at 25 steps with Euler a. Rendering the same seed and prompt at 45 steps for the final produces a noticeably different image. Why?

A. The extra steps revealed fine detail that was always latent in the noise, and that added detail changed how the composition reads at full size.

B. Euler a is ancestral: it adds fresh noise at each step, so it never converges and a different step count follows a different path.

C. The seed only sets the initial noise, so changing any other parameter at all will produce an essentially unrelated image from that starting point.

D. The scheduler re-spaces the noise timetable whenever the total step count changes, and this affects every sampler in the list to the same degree.

The answer and why: p. 384

Lesson 8 · 8 min

Making a big model fit a small card

THE ONE THING TO KEEP

Quantisation and offloading trade quality or speed for memory in known amounts — fp8 costs almost nothing, Q4 costs a little detail, and CPU offload costs seconds rather than quality.

The model does not fit. Now what?

FLUX dev in full precision is about 23 GB of weights. Your card has 8. There are three honest options: use a smaller model, shrink the weights, or keep most of them somewhere else. The second and third are worth understanding because the trade-offs are specific rather than vague.

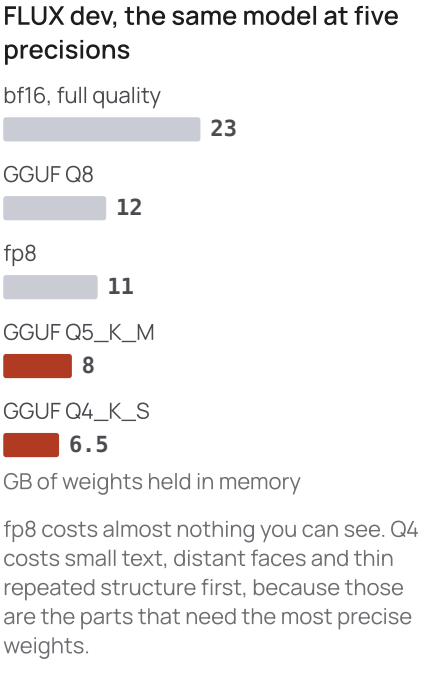
Shrinking the weights

Weights are numbers, and precision is how many bits each number gets.

- **bf16/fp16** — 16 bits. The normal full-quality format. FLUX dev: ~23 GB.
- **fp8** — 8 bits. Halves it to ~11 GB. Quality difference is genuinely hard to see in side-by-side tests; fine detail in skin and foliage shifts slightly, composition does not.
- **GGUF Q8** — ~12 GB, quality near fp8.
- **GGUF Q5_K_M** — ~8 GB.
- **GGUF Q4_K_S** — ~6.5 GB. Now you can see it: fine text degrades faster, very small faces lose structure, and the model is slightly more likely to ignore a difficult part of the prompt.

The mechanism behind that last point is worth stating rather than just the outcome. Quantisation rounds each weight to a coarser grid. The error per weight is tiny, but it accumulates through the layers that resolve fine spatial structure, so the things that need the most precise weights — small text, distant faces, a thin correctly-shaped hand — degrade first, while broad composition is untouched. This is why a Q4 model looks perfectly good on a landscape and disappoints on a portrait with a sign in the background.

Figure 1.4



GGUF support in ComfyUI comes from a community node pack rather than the core, which is worth knowing because it is one more thing that can break on an update.

Keeping weights elsewhere

Offloading leaves parts of the model in system RAM and moves them to the GPU only when needed.

- **Model CPU offload** moves whole components in and out as they are used. Costs a few seconds per image.
- **Sequential CPU offload** goes layer by layer. It makes almost anything fit almost anything, and it is slow — commonly three to ten times slower.
- **Tiled VAE** splits the decode into overlapping tiles so the final step stops being the memory peak.

Offloading costs time and nothing else. Quantisation costs quality and nothing else. If you generate rarely and want the best picture, offload. If you generate constantly and can accept a hair less fine detail, quantise. Doing both is

normal on an 8 GB card.

A worked configuration

FLUX dev on an 8 GB card, which is a common situation:

1. FLUX dev Q5_K_M GGUF, about 8 GB — too big alone, so:
2. Keep the T5 text encoder on CPU. It is large and used once per prompt, so the transfer cost is paid once rather than per step.
3. Enable tiled VAE decode.
4. Generate at 1024×1024, 20 steps.

Expect roughly 60–90 seconds an image. That is slow enough to change your method — you stop browsing and start planning — but it is a real working setup on a card that officially cannot run the model.

The failure this causes

The combination that catches people is quantisation plus LoRA. A LoRA computes a small correction to specific weights; when those weights have been rounded to a coarse grid, the correction lands slightly off, and effects are weaker or subtly wrong. A character LoRA that looks exactly right on fp16 can be a near-miss on Q4, and the instinct is to raise the LoRA strength, which then over-applies the style while still missing the identity.

If a LoRA is not behaving and you are on a quantised model, test it once at a higher precision before you spend an hour tuning strengths. You will often find the LoRA was fine.

Choosing where to spend the memory

On a constrained card you are dividing a fixed budget between four consumers, and it helps to know their relative appetites:

1. **The denoising network** — the bulk of it, and the part quantisation shrinks.

2. **The text encoder** — for FLUX-class models the T5 encoder is several gigabytes on its own. It runs once per prompt, so it is the best candidate for CPU offload: you pay the transfer once rather than on every step.
3. **Activations during sampling** — scales with resolution and batch size. Halving the batch halves this.
4. **The VAE decode** — one short spike at the end, solved by tiling.

The order to attack them in is 2, 4, 3, 1: offload the encoder, tile the decode, reduce the batch, and only then start rounding the weights that make the picture.

Measure it rather than trusting the forum

Quality claims about quantisation are unusually unreliable online, because people compare a Q4 render against an fp16 render at a *different seed* and describe the difference as loss. It is not; it is a different picture.

The test that actually answers the question takes four minutes. Fix the seed, the prompt and every parameter. Render once at the higher precision and once at the lower. Open both at 100% and look at three places: small background text, the smallest face in frame, and any thin repeated structure such as railings or foliage. If you cannot tell them apart there, the quantised model is free quality for your work, and you should stop paying seconds for precision you cannot see.

BEFORE YOU MOVE ON

Someone running a Q4 quantised model finds their character LoRA gives only a loose resemblance. They raise LoRA strength from 0.8 to 1.2 and the style becomes heavy-handed while the face still misses. What is happening?

A. The LoRA was trained at too low a rank to carry a specific identity, and no amount of added strength can make up for capacity that was never there.

B. Strength above 1.0 is outside the valid range and saturates the adapter, so any value above 1.0 will degrade the output no matter what else is true.

C. Quantisation rounded the base weights the correction is computed against, so it lands off-target and strength amplifies the error too.

D. The quantised build uses a different architecture, so only some of the LoRA's keys match and the remainder are skipped without any warning being shown.

The answer and why: p. 385

Lesson 9 · 7 min

Every platform's filter is different, and that is a production risk

THE ONE THING TO KEEP

A refusal is produced by a keyword layer, an output classifier and an account-level policy stacked together, all tuned for the platform's liability rather than your brief, which is why the same prompt passes on one service and fails on another.

Three filters, not one

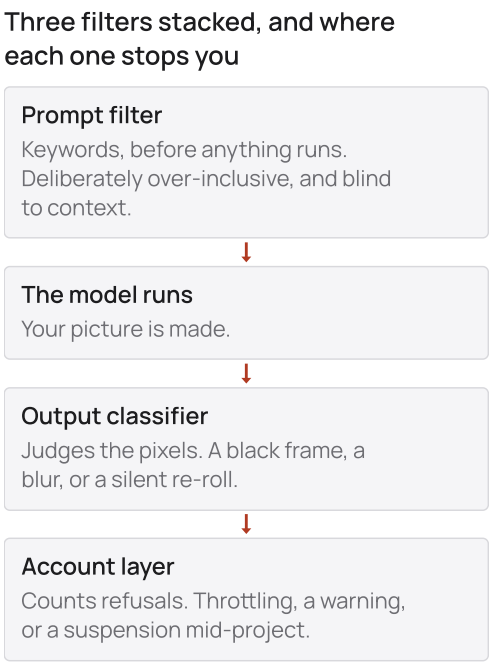
When a hosted service refuses to make your picture, it is rarely a single check. Three layers are usually stacked.

The prompt filter looks at your text before anything runs. It is a keyword and pattern layer, deliberately over-inclusive, and it does not understand context. "Shoot the product against a white wall", "kill the overhead lights", "a child's birthday party", "she is bleeding out from the ears" in a medical illustration brief — all of these have been refused by mainstream services at some point. The refusal is fast and usually says something vague about policy.

The output classifier looks at the generated image before returning it. This one you notice as a black frame, a blur, or a silent re-roll. It catches things the prompt filter could not predict, which is why an innocent prompt sometimes fails on one seed and passes on the next: the classifier is judging the actual pixels, and the model produced something borderline.

The account layer counts your refusals. Several services throttle, warn or suspend after repeated blocks. This matters more than the individual refusal, because a legitimate brief that trips the filter twenty times in an afternoon can cost you the account you are delivering from.

Figure 1.5



Why this is a production problem

A filter is a business decision about the platform's exposure, not a judgement about your work. That has three consequences on a deadline.

1. **It changes without notice.** A prompt that worked in March fails in September because policy tightened after an incident. Your archived workflow is not reproducible on a hosted service in the way a local one is.
2. **It differs across services.** Anatomy for a medical illustration, a historical war photograph, a horror film poster, a bruise for a domestic-violence charity campaign: each of these passes somewhere and fails somewhere else.
3. **Named people are blocked almost everywhere.** Most major services block prompts naming living public figures, and several block historical ones too. This is the filter behaving as designed, and it is not

going to be argued with.

What to do about it

Do not try to defeat a filter. Leaving aside that it usually violates the terms you agreed to, evading a safety system is the fastest way to lose an account mid-project, and "my generator got suspended" is not a thing you can say to a client.

Do the boring things instead:

- **Rephrase clinically.** "Anatomical cross-section, medical textbook illustration" clears filters that "cut open" does not, and describes the brief more accurately anyway.
- **Split the image.** Generate the difficult element separately or photograph it, and composite. Most "impossible" briefs are one small region.
- **Keep a fallback route.** If the work is sensitive — health, conflict reporting, harm-reduction campaigns — plan for open weights on your own machine from the start, where the only filter is the one you choose to run.
- **Check the filter early.** If a campaign concept depends on imagery you suspect is borderline, test it in the first hour, not the night before delivery.

The limitation to be honest about

Local open weights are not filter-free either, and pretending otherwise is a disservice. Most open image models were trained on datasets that had explicit material filtered out, and several have an additional safety checker you can disable. Removing the checker does not restore what was never trained in — the model is simply weak at the categories the dataset excluded, and will produce distorted anatomy rather than the thing you asked for.

And the responsibilities do not disappear with the filter. Non-consensual sexual imagery, sexual content involving minors, and fabricated imagery of real identifiable people are illegal or actionable in most jurisdictions

regardless of what software produced them. A local model removes a platform's guardrail. It does not remove the law, and it certainly does not remove your name from the invoice.

The false-positive tax, in numbers

It is worth having a sense of scale. On a difficult-but-legitimate brief — trauma, conflict, medical, anything involving injury — expect somewhere between one in five and one in three attempts to be refused or silently re-rolled on a mainstream hosted service. That is not a defect report; it is the design point, because the cost of letting through one genuinely harmful image is, for the provider, much higher than the cost of annoying you.

Build that into your estimate. If a brief has four sensitive frames and you are on a hosted service, budget the time as though it has six, and tell the client early that the imagery may need to be shot or composited instead. The version of this conversation you have on day one is short. The version you have at 9pm the night before delivery is not.

Write the refusals down

Keep a short log of what was refused and where. Two lines per incident: the prompt, and the service. Within a month you will have a map of which service tolerates which subject matter, and that map is worth more than any general advice about which provider is "best", because it is measured on the work you actually do rather than on somebody else's.

It also gives you something to show a client when you have to explain why a concept needs rethinking, which is a far better conversation than an unsupported assertion that the tool will not do it.

BEFORE YOU MOVE ON

A charity campaign about domestic violence needs an image showing bruising. Two hosted services refuse it. What is the professionally sound response?

A. Rephrase clinically, and if that fails, plan the sensitive element as a local render or a photograph to composite.

B. Split the description into fragments that each pass the keyword filter on their own, then reassemble the full scene across a series of separate attempts.

C. Accept that this imagery is simply not available from generative tools at present, and go back to the client to change the creative concept entirely.

D. Move to whichever service has the most permissive published policy, since strictness of filtering is the main thing that distinguishes one provider from another.

The answer and why: p. 385

Lesson 10 · 7 min

The setup that still works in six months

THE ONE THING TO KEEP

Diffusion output depends on the exact version of the interface, the model hash and the sampler implementation, so recording those four or five facts with each approved image is what makes a client revision six months later a ten-minute job.

The call you will get

A client approves an image in March. In September they want the same scene with the bottle turned slightly. You open your folder, find the prompt, and generate. It is a different picture.

Nothing is broken. Between March and September you updated the interface, a sampler implementation changed a rounding detail, a node pack bumped its defaults, and the checkpoint you used was re-uploaded with the same name and different weights. Every one of these is common, and any one of them is enough.

The five facts that make an image reproducible

Everything else is noise. These are the ones that actually matter:

1. **Model hash**, not model name. Names are reused constantly. The hash is in the interface's model list and in the generation metadata.
2. **Interface version**, as a git commit, not "latest".
3. **Sampler and scheduler names, plus steps, CFG and seed.**
4. **Every LoRA with its filename, hash and strength.**
5. **Resolution and any upscaler used.**

The interface already writes most of this into the PNG metadata, which is a genuine gift and one most people throw away by exporting to JPEG for the client and deleting the original. Keep the PNG.

Pinning, concretely

For anything you are paid for, treat the setup as a dependency you control.

```
# in your interface folder
git rev-parse HEAD > ENVIRONMENT.txt          # exact commit
pip freeze >> ENVIRONMENT.txt                 # exact package
versions
python --version >> ENVIRONMENT.txt
```

Then, when a project is approved, copy `ENVIRONMENT.txt` into the project folder next to the approved PNGs. It is four lines of shell and it converts "I cannot reproduce this" into "check out that commit".

Do not run `git pull` in the middle of a project. Update between projects, deliberately, and re-run a known image afterwards to see whether anything moved. This feels excessive until the first time it saves you.

A folder structure that survives

```
client-name/
  2026-03-brief.md              # the spec, signed off
  refs/                         # the reference pack
  ENVIRONMENT.txt
  models.txt                    # names + hashes of every model
used
  gen/
    v01/                        # raw PNGs with metadata intact
    v02/
  work/                         # layered editor files
  deliver/
    hero-1200x1500-v03.jpg
```

The two rules that make it work: raw generations never get edited in place, and the delivery folder only ever contains files you have actually sent.

The limit of all this

Reproducibility has a hard ceiling on hosted services, and it is worth being clear-eyed. When you call an API, the provider can update the model behind the same name, change a hidden prompt-rewriting step, or retire the version entirely. Several have done all three. Your seed and parameters are recorded faithfully and still produce a different picture, and there is nothing in your folder that can fix it.

This is the strongest practical argument for keeping open weights locally for work that may need revisiting: a checkpoint file on your disk cannot be updated out from under you. It is also an argument for a specific, cheap habit — when work is approved, archive the layered editor file, not just the prompt. A PSD or XCF with the generated plate on its own layer is reproducible by definition, because the pixels are in it.

Keep the file. Prompts are notes; files are assets.

Back it up, because the disk is the single point of failure

Everything above assumes the files still exist. A working image setup is unusual in that most of its bulk — checkpoints, LoRAs, control models — is re-downloadable and therefore not worth backing up, while a tiny fraction of it is irreplaceable.

Back up the small part:

- The raw generated PNGs with their metadata.
- The layered editor files.
- Workflow graphs, `ENVIRONMENT.txt`, `models.txt`, the brief.

That is usually a few hundred megabytes per project, which fits anywhere. Excluding the model folders from your backup is the difference between a five-minute sync and an overnight one, and it is the reason people stop backing up at all.

One exception is worth carving out. If a project depended on a community checkpoint or LoRA, archive that file too. Uploads are deleted, accounts close, and a model that a client's approved campaign depends on can simply stop existing. A 6 GB file is cheap insurance against a deliverable you cannot ever regenerate.

What "reproducible" honestly means here

Even with everything recorded, exact bit-for-bit reproduction across different hardware is not guaranteed. Floating-point arithmetic is not perfectly associative, GPU kernels differ between vendors and driver versions, and two machines can follow the same seed to visually identical but numerically different images.

For working purposes that is fine: you want the same picture, not the same bytes. Be aware of it when somebody insists a seed produced a different result on their machine — they may be right, and neither of you has done anything wrong.

BEFORE YOU MOVE ON

An approved image cannot be reproduced six months later despite identical prompt, seed, sampler, steps and CFG. Which recorded fact would most likely have explained it?

A. The random number generator differs between CPU and GPU seeding modes, which changes the initial noise even when the seed number is unchanged.

B. The model hash, because checkpoints are often re-uploaded under the same filename with different weights.

C. The original output resolution, because generating at a different size gives the sampler a differently shaped starting latent and therefore a different picture.

D. The negative prompt, which a number of interfaces leave out of the saved metadata block unless the option is explicitly enabled.

The answer and why: p. 386

Lesson 11 · 8 min

You are paying to make decisions you could make cheaply

THE ONE THING TO KEEP

Choose the frame on cheap low-step renders and spend full quality only on the winner; after about six full-quality attempts the remaining distance is editing work, not generation.

Where the money actually goes

A freelancer buys a month of credits, spends two days on one hero image, runs out, and has produced nothing usable. This is the standard failure and it is not about the price of the plan. It is about rendering at final quality while still

deciding what the image is.

Almost every compositional decision — where the subject sits, which of four poses reads best, whether the horizon is high or low, which of three colour moods works — is visible at 512 pixels and 12 steps. That render costs a fraction of a full-quality one. Making those decisions at full quality is like printing every draft of a document on photo paper.

The ladder

1. **Explore small and cheap.** Batches of four to eight at low resolution and low step count. Look for a composition, not for detail. Note the seed of anything promising.
2. **Re-render the winner properly.** Same seed, same prompt, full resolution and step count. The composition transfers. The texture and detail arrive now.
3. **Inpaint the defects**, one region at a time, at full resolution.
4. **Upscale once**, at the very end.

Nine out of ten renders happen at stage one, and stage one is where the cost per render is lowest. That is the whole trick.

Step count is where people burn money out of superstition

Sampler steps cost linearly and stop buying you anything fairly early. With modern samplers, most of the image is settled by 20-30 steps; beyond that you are paying for changes you cannot see. Some distilled models are designed for four to eight steps and get worse if you push them higher.

People run 80 steps because a tutorial in 2022 said so, about a different sampler on a different model. Run the same seed at 15, 25, 40 and 80 once, for your model, and then stop guessing forever.

Guidance scale has the same shape. Too low and the image ignores the prompt; too high and colours blow out and the image goes crunchy. The sweet range is narrower than the slider suggests.

Batch of four, not one at a time

Generate four variants at once rather than one, four times. Two reasons. You judge better by comparison than in isolation — a single image looks fine until you see a better one next to it. And on most hosted tools a batch is cheaper per image than separate requests, because the setup cost is shared.

Keep a log, or you will lose the good one

For anything you might return to: prompt, negative prompt, model and version, seed, steps, guidance, sampler, and the file. A plain text file next to the images is enough.

A prompt on its own is not reproducible. The same words on a different model, or the same model at a different guidance value, give you a different picture. People post prompts online as if they were recipes; without the rest of the settings they are a list of ingredients with no quantities.

Local versus hosted, roughly

Hosted generation sits in the region of a few cents an image on most platforms, so a thousand images is tens of dollars. Running locally is your electricity plus a graphics card you may not own, and a used card capable of comfortable SDXL work has cost a few hundred dollars for a while now.

Break-even is somewhere in the low thousands of images, so for most people hosted is the right answer financially, and local is the right answer for control and privacy rather than for cost.

If you have no money at all, the free path is complete: free daily allowances on several hosted tools, free GPU hours on Colab and Kaggle, and public Spaces on Hugging Face. Do your exploration on the free daily allowance, and spend paid credits only on final renders. That habit alone multiplies what a small budget buys.

And on a preset platform, compare model prices before committing a batch — as lesson two showed, the cheap model and the premium model within one platform can differ by seven or eight times for output nobody can distinguish

at the size it will be used.

Know when to stop generating

If six full-quality attempts at the same thing have not produced it, a seventh will not either. The model has shown you its distribution. The remaining distance is editing work — inpainting, compositing, a tone pass — and doing that is faster and cheaper than continuing to roll.

Set a number before you start. When you hit it, move to the editor.

Today: find the steps setting in whatever you use, run one seed at four different step counts, and see where the improvement stops for your model. Most people discover they have been paying double.

BEFORE YOU MOVE ON

A freelancer exhausts a month's credit allowance in two days on a single hero image and still does not have a usable frame. What is the most useful correction?

A. The plan is too small for professional work and they should move to the next tier up.

B. The composition should have been chosen from batches of cheap, small, low-step renders before anything was rendered at final quality, and after roughly six full-quality attempts the remaining distance is editing work rather than generation.

C. They should move to running an open model locally, where images cost nothing per generation.

D. A higher step count would have reached an acceptable image in fewer attempts.

The answer and why: p. 386

CASE STUDY · MODULE 1

Two hundred assets, and a licence nobody had read

An illustrative case, built from documented patterns.

A two-person branding studio in Indore, eleven weeks into a retainer for a regional dairy co-operative.

Farhan and Divya run a studio out of a converted flat. Between them they had delivered two hundred and six images for the co-operative over eleven weeks: packaging mock-ups, market-stall backgrounds, a series of illustrated farmer portraits for the rebranded milk cartons, and a set of forty social cards. All of it had been generated locally, on a second-hand card Farhan had bought precisely so that per-image cost would stop being a line in their quotes.

The model was FLUX.1 [dev]. They had chosen it in week one because it held type better than anything else they could run, the samples on the model page were the best they had seen, and the download was free and took twenty minutes. Nobody read the licence file that came down with the weights. It was sitting in the folder the whole time.

In week eleven the co-operative's new marketing head asked a reasonable question: for the packaging artwork, which is going to a printer and onto half a million cartons, what are we licensed for? Divya opened the model card to copy a line into the email and found that FLUX.1 [dev] is released under a non-commercial licence. The weights may be used for research and personal work. Generating output that a client pays for is not that.

Two things were immediately clear and one was not. The first: this was a breach of the model licence, and it was entirely separate from any question about who owns the pictures. The second: it was not the client's fault and the client had no exposure they knew about. The third, the unclear one, was what to do.

The options were not equal.

Say nothing. Nobody had noticed in eleven weeks. The licence is not policed by anybody who would plausibly look at a dairy co-operative in Madhya Pradesh. The studio's reputation, their retainer and their next quarter all survive intact. The cost is that Divya would have to answer the marketing head's direct question with something that was not true, in writing, and then live with that sentence existing in an inbox for as long as the account did.

Rebuild everything. Re-generate two hundred and six images on Apache-licensed weights — FLUX.1 [schnell] or Qwen-Image, both of which permit commercial use — then repair, composite and grade them all again. Farhan estimated it at nine days of work they would not be paid for, and it would push two deliverables past dates the co-operative had already booked print slots against.

Rebuild only what is exposed. The packaging artwork and anything going to print is the part with real exposure and volume behind it. The social cards are ephemeral. That is about thirty images rather than two hundred, and it is defensible if you can say where the line was drawn and why.

Buy the licence. Black Forest Labs sells a commercial licence for the dev weights. The studio had never priced it because they had never thought they needed to.

Divya's position was that the third option was a rationalisation dressed as a plan: the licence does not distinguish between a carton and a social card, and drawing a line by commercial risk is deciding which breaches matter. Farhan's position was that nine unpaid days would take the studio's cash below the point where they could pay the rent in April, and that a rule you cannot afford to follow completely is still worth following partly.

The argument took a day. It was not really about the licence by the end of it. It was about whether a two-person studio that does not check terms before it starts is a studio a client should hire, and whether either of them wanted to keep working in a way that depended on nobody asking.

YOUR ANSWERS

1. Why did buying the commercial licence not, on its own, settle the question?

2. Farhan proposed rebuilding only the print work. What is the strongest argument for that position, and why did it lose?

3. What is the difference between the question the marketing head asked and the question about who owns the images?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 62.

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

They priced the commercial licence first, and it was affordable — less than one of the nine unpaid days would have cost them. They bought it, which made the past eleven weeks compliant going forward but did not retroactively license what had already been made, so Divya wrote to the marketing head the same afternoon, said plainly what had happened, said it was now licensed, and offered to regenerate the packaging artwork at no charge if the co-operative's own lawyers wanted a clean provenance. The co-operative asked for the packaging artwork to be regenerated, and nothing else. It took three days. The studio kept the account and added one step to their project template: a line in the brief naming the model, its licence and the date the licence page was read, filled in before the first render. Two months later that line stopped a different job from starting on a community merge whose ancestry nobody could establish.

The questions, discussed

1. Why did buying the commercial licence not, on its own, settle the question?

A licence bought in week eleven governs use from week eleven. The two hundred images already delivered were generated while the studio was not licensed to use the weights commercially, and no later purchase changes what happened. That is why the disclosure still had to be made: the fix was for the future, and the client was the one carrying the artwork.

2. Farhan proposed rebuilding only the print work. What is the strongest argument for that position, and why did it lose?

The strongest argument is proportionality: the packaging carries volume, permanence and a printer's audit trail, so it is where a claim would actually land, and spending nine unpaid days on ephemeral social cards buys almost no reduction in real exposure. It lost because the licence does not draw that line, so choosing it means deciding privately which breaches count — which is a position the studio could not have explained to the client if asked.

3. What is the difference between the question the marketing head asked and the question about who owns the images?

They are separate. Being licensed to use a model is a term you can look up on the model card in two minutes and which has a definite answer. Whether the output is protectable, and by whom, is unsettled and differs by country — India deems the person who made the arrangements to be the author, the United States requires human authorship. The studio failed the easy question, not the hard one.

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. 380. If two go wrong, re-read the module before the exam.

1 You download FLUX.1 [dev], run it on your own card, and bill a client for the images. Which statement is accurate?

A. It breaches the model licence, whatever the position on who owns the pictures.

B. It is fine as long as you tell the client the images were made with AI.

C. It depends on whether your country grants copyright in computer-generated works to the arranger.

D. It is fine, because the weights were free to download and you generated the images yourself.

2 A client approves a bottle shot made on a preset platform, then asks for the light to come from behind the glass instead of the front. Why does rewriting the prompt not reach it?

A. Lighting terms only work on open weights, not on hosted services.

B. The prompt would have to exceed the model's token window to carry that instruction.

C. The platform charges more credits for a revision than for a first generation.

D. The lighting decision lives inside the preset and is applied after your words.

3 Your first generation after switching checkpoints takes four minutes; every later one on the same model takes fifteen seconds. What is the likeliest cause?

A. The VAE decodes at full resolution on the first pass and at half on later ones.

B. System RAM is short, so loading the checkpoint is swapping to disk.

C. The sampler is ancestral and injects fresh noise only on the first run.

D. The card is thermally throttling and recovers once the fans have spun up properly.

4 Generation reaches one hundred per cent and then fails with CUDA out of memory. Which fix addresses the actual peak?

A. Raise the batch size so the fixed setup cost is shared across more images.

B. Lower the step count, since every step holds its own set of activations in memory.

C. Turn on tiled VAE, because the decode is the peak rather than the sampling.

D. Switch to a non-ancestral sampler so memory is not reallocated on every step.

5 A character LoRA that was exact on fp16 is a near-miss on a Q4 model. What is happening, and what should you not do about it?

A. The checkpoint shipped a pruned VAE, so swap the VAE before touching anything else.

B. The LoRA was built for a different base checkpoint, so it should be retrained from scratch against the quantised file.

C. Q4 drops the text encoder, so the fix is to raise the LoRA's text-encoder weight.

D. Rounded weights land the LoRA's correction slightly off; do not simply raise its strength.

6 A client approves an image in March and asks for a small change in September. The same prompt and seed give a different picture. Which record would have prevented it?

A. The model hash, the interface commit, and every LoRA with its strength.

B. The negative prompt, which is the setting people most often forget to write down.

C. The model's name and the date you downloaded it.

D. A JPEG of the approved image, kept in the delivery folder.

2

MODULE 2

From a request to a specification you can generate against

Convert a one-line request into a written image specification — final pixel dimensions, the native generation ratio and the crop-safe area inside it, colour and brand constraints, a purpose-labelled reference pack and an explicit forbidden list — and state which elements will have to be photographed or composited rather than generated.

12 Start at the placement, not the picture 69
13 The ratios the model was trained on, and what happens outside them 72
14 Designing for the crop you will not control 77
15 Six references, each with a job 81
16 Write down what must not appear 84
17 A hex value is not something a diffusion model can hit 88
18 One hero or twelve pieces is a different job 91
19 The frames where a phone camera wins 95
20 The one page that ends the revision argument 99
21 What you can sell, and what you must disclose 104

Case study: The coaching centre that wanted a campus it did not have 109
Module 2 test 114

Lesson 12 · 8 min

Start at the placement, not the picture

THE ONE THING TO KEEP

The final placement fixes the dimensions, the crop, the amount of detail that will ever be visible and the file format, and every one of those decisions is cheaper to make before generating than after.

"Something for the launch"

That is the brief you will actually receive. It contains no dimensions, no placement, no ratio and no idea of whether the thing will be seen at 400 pixels wide on a phone or at two metres on a wall. Generating against it wastes an afternoon, because the first question at review will be "can we get it in a square for Instagram too", and the answer will be no.

The fix is one question asked first: **where does this appear?** Everything else follows from the answer.

The numbers you will actually meet

Keep these somewhere. They change slowly and they decide your generation size.

- **Instagram feed portrait** — 1080 × 1350 (4:5). The single most common social deliverable.
- **Instagram/TikTok story or reel cover** — 1080 × 1920 (9:16).
- **Open Graph card** for a link preview on most platforms — 1200 × 630 (roughly 1.91:1). Cropped hard on both sides on some surfaces.
- **YouTube thumbnail** — 1280 × 720 (16:9), and it will be displayed as small as 246 × 138.
- **Website hero** — commonly 2560 wide for a desktop banner, with the height decided by the section, and a separate mobile crop.
- **A4 page at 300 dpi** — 2480 × 3508. A5 is 1748 × 2480.

- **A3 poster at 300 dpi** — 3508 × 4961.
- **Roll-up banner, 850 × 2000 mm** — at 150 dpi that is 5020 × 11811, which no generator produces directly and which you reach by upscaling a correctly proportioned render.

Two of these deserve a note. The YouTube thumbnail is a 1280-pixel image that most people see at a fifth of that size, so detail below a certain scale is wasted effort and contrast is everything. And large-format print does not need 300 dpi: a roll-up banner viewed from two metres is perfectly sharp at 100–150 dpi, and a billboard at 30 metres is fine at 15. People generate uselessly enormous files because they learned one number and applied it everywhere.

What the placement tells you beyond size

How long it is looked at. A story frame gets under two seconds. A poster on a wall gets thirty. That difference decides how many elements the composition can carry, and it is a stronger constraint than any style choice.

Where the type goes. If a headline will sit across the top third, that third needs to be quiet — a plain sky, a wall, a gradient — and you should generate it that way rather than blurring it afterwards. Asking for "negative space in the upper third for a headline" in the prompt works surprisingly well and costs nothing.

What the format has to be. A photographic hero delivers as JPEG or WebP. A product cut-out for a shop listing needs a transparent PNG, which no diffusion model outputs directly — it comes from generating on a plain background and cutting it out afterwards. Discovering this at delivery is an unpleasant hour.

Whether it must tile or repeat. A background pattern has a constraint no amount of prompting fixes casually, and it needs a seamless-tiling workflow chosen at the start.

Ask these four questions

Before generating anything, you want four answers in writing. They take one message to obtain:

1. What are the final pixel dimensions, and are there other sizes it must also work in?
2. Will type sit over it, and where?
3. Print or screen? If print, what process and what size?
4. Does anything real have to be in it — a specific product, a named person, a building?

That fourth question is the one that saves the most time, and the next lessons in this block are largely about what to do when the answer is yes.

The limit worth stating plainly

A single generated image cannot serve every placement. A 1080 × 1350 portrait crops badly to 1200 × 630, and stretching it is visible. The professional answer is to generate wider than you need and crop down, which the next lesson covers, or to generate each placement separately from a shared reference.

What you should not do is promise one image will cover six placements before checking the ratios. It is the commonest small failure in this work, it is discovered at the worst possible moment, and it is avoided by thirty seconds of arithmetic at the start.

BEFORE YOU MOVE ON

A client asks for a launch image for "socials and the website". What is the most useful first action?

A. Ask for the exact placements and their pixel sizes, since ratio, detail and type placement all follow from them.

B. Generate a square image, because a square is the format that crops acceptably to the widest range of social and web placements without visible distortion.

C. Generate at the highest resolution the model supports so that every placement can be produced by downscaling from a single large master file later.

D. Begin with the visual concept and settle the technical dimensions once the client has approved a direction they are happy with.

The answer and why: p. 388

Lesson 13 · 8 min

The ratios the model was trained on, and what happens outside them

THE ONE THING TO KEEP

Models are trained at a specific set of resolutions, and generating far outside those buckets produces duplicated subjects and stretched anatomy, so you pick the nearest trained ratio and crop to the delivery size afterwards.

Why a tall image grows a second head

Ask an SDXL model for 1024 × 2048 and you will often get two torsos stacked, or a figure with an extra set of legs. This is not the model being stupid. It is a direct consequence of how it learned.

Training images are grouped into **buckets** — a fixed set of resolutions with roughly constant total pixel count. SDXL's bucket list is built around 1,048,576 pixels, which is why the familiar sizes are:

- 1024 × 1024 (1:1)
- 1152 × 896 and 896 × 1152 (roughly 9:7)
- 1216 × 832 and 832 × 1216 (roughly 3:2)
- 1344 × 768 and 768 × 1344 (roughly 7:4)
- 1536 × 640 and 640 × 1536 (12:5)

Inside those, the model has seen thousands of examples of how a person fits the frame. Outside them, it has seen almost none, and its behaviour is an extrapolation. The specific failure — repeated subjects — happens because the network's receptive field covers a certain span of the latent; stretch the canvas well beyond what it saw and two regions independently decide they are each looking at the centre of a portrait. You get two centres.

SD 1.5 has the same structure around 512 × 512 and degrades much faster outside it; anything past about 768 in either dimension repeats reliably. FLUX and other newer architectures are considerably more tolerant and will hold together at ratios SDXL cannot, but they still have a comfortable range and still degrade outside it.

The working rule

Generate at the nearest native bucket. Crop or extend to the delivery size afterwards.

Two worked examples:

- You need 1080 × 1350 (4:5) for Instagram. The nearest SDXL bucket is 896 × 1152 (7:9, which is 0.778 against your 0.8). Generate there, upscale to 1080 × 1389, crop 39 pixels. Nothing distorts.
- You need 1200 × 630 (1.905:1). The nearest bucket is 1344 × 768 (1.75:1). Generate there and **outpaint** the sides rather than cropping the top and bottom, because cropping a 1.75 frame to 1.9 throws away composition you

wanted. Outpainting is covered later in the course, and this is the commonest reason to reach for it.

For a very wide or very tall final format — a roll-up banner at 1:2.35, a web skyscraper ad — do not try to generate it in one go. Generate the interesting part at a native ratio and extend outwards in steps. The result is coherent; the single-shot attempt is a stack of duplicated torsos.

Figure 2.1

Where SDXL composes, and where the work is delivered

Generate here — the trained buckets

- 1024 × 1024, square
- 1152 × 896, about 9:7
- 1216 × 832, about 3:2
- 1344 × 768, about 7:4
- 1536 × 640, 12:5 and the widest that holds

Deliver here — crop or extend afterwards

- 1080 × 1350 feed portrait: upscale, crop 39 px
- 1200 × 630 link card: outpaint the sides
- 1280 × 720 thumbnail: crop from 1344 × 768
- 2480 × 3508 A4 at 300 dpi: enlarge separately
- 5020 × 11811 roll-up: extend outwards in steps

Every bucket on the left is about 1,048,576 pixels. Outside them the network's receptive field covers more canvas than it ever saw, two regions each decide they are the centre of a portrait, and you get two torsos.

Checking a model rather than trusting a table

Bucket lists differ between model families and the table above is only for SDXL. A four-minute test tells you any model's real range: fix the seed and prompt, generate a simple single-subject portrait at 1:1, 4:5, 3:2, 16:9 and 2:1, and look at which ones hold. Write the answer next to the model in your notes. This is the sort of thing everybody re-learns from forum posts every few months and nobody measures once.

What this does not fix

Bucketing explains duplicated subjects at extreme ratios. It does not explain a duplicated subject at 1:1, which has a different cause — usually a prompt that describes a scene wide enough that the model reasonably renders several people, or a composition control pushing in two directions.

And there is an honest limit to the "generate native, crop later" rule. Cropping changes the composition the model chose. A frame that was well balanced at 3:2 can lose its subject's breathing room when taken to 4:5, and the fix is not a better crop but generating with the final crop in mind — which is what the next lesson is about. Ratio arithmetic gets you a frame with no duplicated limbs. It does not get you a frame that still works after the crop.

Resolution is a separate question from ratio

People conflate the two and then generate a 2048 × 2048 image expecting more detail. Past the trained pixel count you do not get a more detailed picture; you get the same failure as an extreme ratio, because the canvas is again larger than anything the model composed for. A 2048-square SDXL render very often contains two horizons, two suns, or a subject assembled from two overlapping attempts.

The route to a large image is therefore always the same: **generate at native size, then enlarge in a separate step**. An upscaler — an ESRGAN-family model, which is free, tiny and needs no GPU worth speaking of — takes a good

1024 frame to 4096 without inventing a second horizon, because it is doing a different job. The finishing block covers what upscalers add and what they fabricate.

There is one partial exception. Some newer models handle larger canvases far better than SDXL does, and a few are trained at 1536 or 2048 natively. The test from the section above tells you which you have in four minutes. Do not assume from a version number.

BEFORE YOU MOVE ON

An SDXL render at 832 × 1600 produces a figure with two sets of shoulders. What is the correct response?

A. Raise the guidance scale so the model adheres more strictly to the single-subject description given in the prompt.

B. Add "one person, single subject, solo" to the prompt and "two people, duplicate, twins" to the negative prompt in order to suppress the extra figure.

C. Generate at 832 × 1600 with a substantially higher step count, since the duplication is an artefact of the image not having fully converged at the number of steps currently set.

D. Generate at a native bucket such as 832 × 1216, then outpaint upward in steps to the final height.

The answer and why: p. 389

Lesson 14 · 8 min

Designing for the crop you will not control

THE ONE THING TO KEEP

Plan the frame as concentric zones — a core that survives every crop, a middle that survives most, and an outer band that exists to be thrown away — because the image will be re-cropped by people and platforms who never see your original.

Your image will be cropped by strangers

You deliver a beautiful 4:5 frame. It appears as a 1:1 thumbnail in a feed, as a 16:9 card when shared to a messaging app, as a circular avatar somewhere, and with a 200-pixel strip of it behind a headline on the website. You are not consulted about any of this.

The answer used by broadcast and print for decades is **safe areas**, and it transfers directly.

Three zones

Think of the frame as three nested regions.

The core, the central square or thereabouts. Everything the image *means* lives here: the subject's face, the product, the gesture that carries the idea. This survives every crop anyone will apply.

The support band, out to roughly 85% of the frame. Context that makes the image good but not essential — the table the product sits on, the second figure, the environment. Some crops keep it, some do not.

The sacrificial edge, the outer 10–15%. Texture, gradient, background. Designed to be cut. Never put anything here that must be seen.

Figure 2.2

Three nested zones, from the middle outwards

The core — the central square, roughly
Face, product, the gesture. Survives every crop.

The support band — out to about 85%
Context that helps. Some crops keep it, some do not.

The sacrificial edge — the outer 10 to 15%
Texture and background. Designed to be thrown away.

Eyes forty pixels from the top edge are not bad luck. A 1:1 crop of a 4:5 frame takes them, and the composition was never crop-safe.

If your subject's eyes are 40 pixels from the top edge, a 1:1 crop of a 4:5 frame will cut them. That is not bad luck; it is a composition that was never crop-safe.

Getting a generator to leave you room

Generators pull toward filling the frame — the training data is full of well-composed photographs where the subject dominates. To get usable margin, ask for it explicitly, and in the language photography uses:

- "wide shot, subject small in frame, generous negative space above"
- "centred composition, plain background, copy space at the top"
- "full body with headroom, shot on a 35mm lens"

Lens language does real work here. "85mm portrait" produces a tighter, more compressed frame; "24mm wide" gives you room and a more environmental feel. These are more reliable levers than the word "space", because the model has seen the lens described in thousands of captions alongside the framing it produces.

The other approach is to generate at a wider ratio than you need and treat the extra as margin by design. Render 1344 × 768 when you need 1200 × 630, and you have 144 pixels of horizontal slack and 138 of vertical to place the crop with, after the fact, where it actually works.

The type zone is part of the composition

If a headline goes across the top third, that third has two requirements: it must be quiet, and it must have enough contrast against white or black type to be legible. A busy sky with bright clouds fails the second even though it is "empty".

Generate for it. "Plain overcast sky occupying the upper third" gives you a place to set type. So does "dark studio background, subject lit from the side, lower two thirds". You are not decorating; you are building a surface that type can sit on, and the finishing block returns to the legibility arithmetic.

Check it before you deliver

Ninety seconds in any editor, including free ones:

1. Open the image in GIMP, Krita or Photopea.
2. Add guides at 1:1, 16:9 and 9:16 centred on the frame.
3. Look at each. Is the subject intact? Is anything important cut? Does the 1:1 still read at 150 pixels?

If a crop fails, you have three options and they are not equal. Re-crop, which is free. Outpaint to add room, which takes a few minutes. Or regenerate, which costs the composition you had approved. Doing this check before the client sees anything means you only ever have the cheap version of the problem.

The limitation

Crop-safe composition costs you the strongest version of the picture. A frame designed to survive five crops is, almost by definition, more centred and more conservative than one composed for a single format. That is a real trade, not a free technique.

So make it deliberately. A hero image that only ever appears in one place should be composed for that place and nothing else. Crop safety is for images that will travel — social assets, stock, anything a client's marketing team will re-use without asking. Applying it everywhere produces a portfolio of safe, centred, slightly dull pictures, and that is its own kind of failure.

BEFORE YOU MOVE ON

A 4:5 social image keeps losing the subject's face when platforms crop it to a square thumbnail. What is the underlying composition error?

A. The image was generated at a non-native aspect ratio, so the model placed the subject inconsistently within the frame.

B. Important content was placed in the band that only survives the original ratio, instead of the central core.

C. The resolution is too low, so the platform's automatic crop cannot identify the subject and defaults to a centred rectangle.

D. The delivered file should have been a square from the beginning, because any image reused across several placements ought to be generated in the most neutral format available.

The answer and why: p. 389

Lesson 15 · 7 min

Six references, each with a job

THE ONE THING TO KEEP

A mood board of vibes cannot be acted on, but six references each labelled with the single thing it is there to demonstrate can be used directly as conditioning and as the basis for sign-off.

The problem with a mood board

A client sends twenty pictures in a shared folder. They are all "the feeling we want". Half are lit differently from the other half, three are illustrations and the rest photographs, and two contain a competitor's product. You generate something that averages them and it satisfies nobody, because nobody agreed on what they were agreeing to.

A reference pack fixes this by attaching a **job** to each image. Not "this is nice" — "this one is here for the light".

The six slots

Six is usually enough, and forcing the limit is part of the method.

1. **Light.** One image that demonstrates the lighting: direction, hardness, colour temperature. This is the single most influential reference and the one people forget.
2. **Palette.** One image whose colours are right, even if its subject is wrong.
3. **Composition.** A frame whose arrangement you want to echo — subject placement, camera height, how much room around things.
4. **Subject or product.** What the thing actually is. For a real product this is a photograph of the real product, and it is non-negotiable.
5. **Texture and finish.** Grain, gloss, matte, film stock, print quality. This decides whether the result looks like a photograph, a render or a scan.

6. **The anti-reference.** One image that is close to the brief and wrong, with a sentence saying why. This is worth more than two of the others, because it converts taste into a statement people can check.

Write one line under each. "This one: the hard sunlight from the left, not the model, not the styling." That sentence is what stops the client saying at review that they never meant the person to look like that.

Why this is not just admin

The labels are how you use the references technically. Later in the course you will be giving images to the model directly rather than describing them, and each slot maps onto a different mechanism:

- The **composition** reference becomes a depth or edge control.
- The **palette** reference becomes a colour-matched starting image at low denoise.
- The **light and texture** references become style adapter inputs, or vocabulary you extract by interrogating them.
- The **product** reference becomes a composite, not a generation.

A pile of twenty unlabelled images cannot be routed this way. Six labelled ones can be, and the routing is most of the craft in the middle of this course.

Where references come from, honestly

Use what is in front of you. Photographs you took, stills from films, museum collections that publish open-licence scans, the client's own archive, free stock libraries. There is no need to be precious about looking at other people's work to describe what you want — that is what art direction has always been.

Two cautions that matter in practice. First, a reference is for *direction*, not for reproduction: taking a living illustrator's picture and conditioning on it heavily enough that the output is recognisably theirs is a problem regardless of what any court eventually decides about training data, and the later block

on rights returns to it. Second, if the client supplies references, ask where they came from. "We found it online" is the beginning of an awkward conversation later, not the end of one.

The one-line test

Before generating, read the pack back as a sentence: "Hard side light like ref 1, the muted greens of ref 2, arranged like ref 3, our actual bottle from ref 4, with the matte film texture of ref 5, and specifically not the plastic studio look of ref 6."

If that sentence is coherent, you have a brief. If it contradicts itself — hard sunlight and soft studio texture — you have found the disagreement now, in a message, rather than at the review where somebody has to lose.

Building the pack when the client sends nothing

Often you get no references at all, and asking for some produces silence. Do not wait. Assemble a pack yourself and send it as a question: "Which of these three directions?"

Three columns of three images each, each column internally consistent, each clearly different from the others. Warm and documentary; cool and graphic; soft and domestic. The client picks one in under a minute, which is a decision they can make, whereas "what do you want it to feel like" is one they usually cannot.

This is the oldest trick in art direction and it works because recognition is much easier than description. You are not asking them to design anything. You are giving them three things to react to, and a reaction is a specification you can build on.

Two practical notes. Make the three genuinely different — three variations on the same idea teaches you nothing. And keep the column they chose, labelled, because it becomes the pack, and the two they rejected become your anti-references with a reason attached.

BEFORE YOU MOVE ON

What makes a labelled six-image reference pack more useful than a twenty-image mood board?

A. Six images can be supplied to the model together as conditioning inputs, whereas twenty would exceed what any image-conditioning mechanism is able to accept at once.

B. Each image names the one attribute it contributes, making it checkable at sign-off and routable to a conditioning method.

C. A smaller set reduces the risk that the generated output will resemble any single source image closely enough to raise a copyright question later on.

D. Twenty references take substantially longer to review with a client, and the time saved in that meeting is the main practical benefit of trimming the set down.

The answer and why: p. 390

Lesson 16 · 7 min

Write down what must not appear

THE ONE THING TO KEEP

A negative prompt is a weak statistical nudge rather than a guarantee, so anything that would be genuinely damaging in the final image belongs on a written checklist that a person verifies before delivery.

The things that get you into trouble

Every brief has an unwritten list of things that must not be in the picture. Nobody says them out loud because they are obvious, right up until one appears in an approved asset that has already gone to print.

Real examples, all of which have happened:

- A generated street scene containing a sign in something that looks like a competitor's typeface and colours.
- A can whose silhouette and red-and-white banding read unmistakably as a trademarked product.
- A crowd scene for a children's charity in which two of the generated children are dressed inappropriately.
- An "Indian family at dinner" that returned every visual cliché of the stock-photo canon, delivered to a client who is Indian.
- A background bookshelf whose spines contain garbled text that happens to spell something unfortunate.

None of these come from bad prompting. They come from a model reproducing what is statistically common in its training data, and from nobody having written down what would be unacceptable.

Why the negative prompt is not the answer

The instinct is to put the forbidden things in the negative prompt. Understand what that actually does: the negative prompt produces a second prediction that the sampler steers *away* from, proportionally to the guidance scale. It biases the outcome. It does not forbid anything.

So "no logos" in the negative reduces the frequency of logos. It does not prevent them. For something that would cost a client a legal letter, a probabilistic reduction is not a control — it is a hope with a technical-sounding name.

There is a second, sharper problem. Negative prompting is unreliable for absences and for composition, and it can actively backfire by reinforcing a concept the model then represents. The related lesson elsewhere in this catalogue on how "without" summons the thing is worth reading alongside this one.

The list, and who checks it

So write it down, in the brief, as items a human verifies on the finished file.

```
MUST NOT APPEAR
- Any legible brand name or logo, including on background
signage
- Alcohol, cigarettes
- Any child under ~12 in frame
- Religious symbols
- Recognisable real landmarks (the client operates in four
countries)
- Text of any kind other than the type we set ourselves
- Hands holding the product (we cannot match skin tone across
markets)
```

Two features of that list matter. It is **specific** — "no brands" is unactionable, "no legible brand name including on background signage" is checkable. And it is **verifiable by looking**, which means it belongs in the pre-delivery QA pass rather than in a prompt.

Doing it at generation time as well

The list is the guarantee; there are still sensible things to do while generating, because they reduce how much repair you need.

- **Make the background simple.** Most accidental logos, text and clutter arrive in busy backgrounds. "Plain wall", "shallow depth of field", "clean studio background" removes whole categories of risk at source.
- **Do not name the thing you do not want.** Describe the scene you do want, positively.
- **Generate the risky element separately.** If the brief needs a hand holding the product and hands are a problem, photograph the hand.
- **Check at 100%.** Garbled text and stray marks are invisible in a thumbnail and obvious on a client's monitor.

The one that is hardest to catch

Stereotype is the item on this list that is easiest to miss and most damaging when missed, because the model is not producing an error — it is producing the average of what its training data associates with your words, and that

average carries whatever the source material carried.

There is no prompt that fixes this reliably. What works is naming specifics rather than categories — an occupation, a place, an age, a piece of clothing — so the model has something to condition on beyond the demographic term. And what works better is a person looking at the output and asking whether they would be comfortable if it were about them. That is a review step, not a setting, and it belongs in the brief so it actually happens.

BEFORE YOU MOVE ON

Why should "no visible brand logos" be a delivery checklist item rather than a negative prompt entry?

A. Because a negative prompt shifts the odds rather than enforcing a rule, and the consequence of one logo slipping through is legal rather than aesthetic.

B. Because negative prompts stop working reliably once the guidance scale is lowered below about 5, which is the range most photographic checkpoints are actually run at.

C. Because logos are usually generated in background regions that the negative prompt has less influence over than it does on the main subject of the frame.

D. Because naming a brand in the negative prompt may itself constitute a use of that trademark, which introduces a risk separate from the one the entry was meant to remove.

The answer and why: p. 390

Lesson 17 · 8 min

A hex value is not something a diffusion model can hit

THE ONE THING TO KEEP

Diffusion models cannot be instructed to produce an exact colour, so brand colour is achieved by generating in a neutral or near palette and then grading or compositing the exact value in afterwards.

#E4002B, and why asking does not work

A brand guideline says the red is #E4002B. You put that in the prompt. You get a red. It is not that red, and it is not the same red twice.

The reason is structural. The prompt goes through a text encoder that turns it into a semantic embedding — a representation of meaning. "#E4002B" is tokenised into fragments that carry no reliable meaning, and even "crimson red" is a region of colour space rather than a coordinate. The model then denoises toward an image consistent with that meaning, with every pixel influenced by lighting, material, white balance and the rest of the scene. There is no path in that architecture by which a specific RGB triple can be requested.

This is worth understanding rather than just accepting, because it tells you where the control actually is: **after generation, in pixel space**, where a colour is just a number you can set.

Four ways to get the exact colour

One: generate neutral, grade after. The most reliable method. Generate the scene with the element in a plain, slightly desaturated version of the colour, then correct it in an editor. In GIMP or Photopea, select the region, and use Curves or Hue-Saturation to move it. For a flat surface, a solid-colour layer in Colour blend mode over a mask lands the value exactly while keeping the original shading.

Two: composite the element. If the brand colour belongs to an object with a defined shape — a packet, a bottle, a chair — put the real thing in. This is the product workflow from later in the course and it solves colour as a side effect.

Three: condition on a colour block. Paint flat shapes of the right colours in GIMP or Krita, then run that image through img2img at low denoise, around 0.3–0.45. The composition and palette survive; the model adds material and light. This is the best method when the brand colour must be distributed across a whole scene rather than on one object.

Four: accept approximate and say so. For background atmosphere and incidental colour, "in the region of the brand palette" is usually fine, and pretending otherwise creates work nobody asked for. Ask which elements are strictly specified. Usually it is two.

The thing that surprises people

Even after you set the exact hex, it will look wrong in context. A pure brand red surrounded by warm generated light reads as orange-shifted; the same red on a cool background reads pink. This is simultaneous contrast and it is a property of human vision, not a colour error.

Which means grading the whole image matters as much as the swatch. If the generated scene is warm and the brand palette is cool, you either grade the scene toward the palette or you have a permanent argument about whether the colour is right. Do that first and match the specific element second.

Print makes it worse, and it is better to know now

If the image is going to print, the brand colour may be defined as a spot ink that no CMYK press can reproduce, let alone your screen. A vivid brand orange or a deep navy frequently shifts visibly when converted. That is a standard print problem rather than a generation problem, and the vector-and-print course in this catalogue covers the conversion properly, but the timing matters here: you want to know before you spend two hours matching a screen colour that the press cannot hit.

A free toolchain that does all of this

None of it needs paid software.

- **GIMP** — Curves, Hue-Saturation, selection by colour, layer blend modes.
- **Krita** — better painting tools for making colour-block conditioning images.
- **Photopea** — runs in a browser, reads and writes PSD, useful when you are on a borrowed machine.
- **Inkscape** — for placing exact flat brand colours as vector shapes over a generated plate, which is how most logos and blocks of brand colour should arrive anyway.

The rule underneath all four methods is the same: **use the generator for light, material and form; use pixel-editing tools for exact values.** Trying to make a diffusion model do arithmetic on colour is spending hours on something a colour picker does in one click.

BEFORE YOU MOVE ON

Why can a diffusion model not reliably produce an exact hex colour on request?

A. Because the text encoder turns the prompt into a semantic embedding, so there is no channel by which a specific numeric value reaches the pixels.

B. Because the VAE decode from latent space to pixels applies a colour transform that shifts values away from whatever the sampler produced during denoising.

C. Because the model's training captions almost never contained hex codes, so more examples of colour-coded captions would eventually make the request work.

D. Because guidance scale controls saturation as well as prompt adherence, so any specified colour drifts as the guidance value changes between generations.

The answer and why: p. 391

Lesson 18 · 7 min

One hero or twelve pieces is a different job

THE ONE THING TO KEEP

A single image is a search problem and a set is a consistency problem, and they need different budgets, different methods and a decision made before any generating starts.

Two jobs that look the same on the brief

"We need an image for the campaign" and "we need the campaign images" differ by one word and by a factor of about four in effort, and not in the direction you would guess.

One hero image is a search. You are looking for a frame that is good enough to carry the whole thing, and the method is breadth: many cheap candidates, a ruthless cull, then deep work on one winner. Budget shape: 70% exploration, 30% finishing.

A set of twelve is a consistency problem. The individual images do not need to be extraordinary; they need to look like each other. One stunning frame among eleven ordinary ones actively hurts the set, because it makes the rest look like mistakes. Budget shape: 30% establishing the system, 20% generating, 50% making them agree.

The mistake is running a set like twelve heroes. You get twelve good, unrelated pictures, and then a very long day trying to reconcile them.

What "the system" means

For a set, the first job is not an image — it is a set of constants you will hold across every frame:

- **Light**: direction, hardness, colour temperature. One choice for all twelve.

- **Lens**: a focal length and a camera height. Mixing a 24mm wide and an 85mm portrait in one set reads as two different shoots.
- **Palette**: three or four colours that recur.
- **Grain and finish**: applied identically at the end, which is the cheapest consistency you can buy.
- **Crop convention**: how much room around the subject.

Fix those, write them down, generate one **anchor image** that embodies all of them, and approve it before making the other eleven. The anchor then serves technically as well as editorially — as a style reference, as a colour-match target, and as the thing you hold up when frame nine drifts.

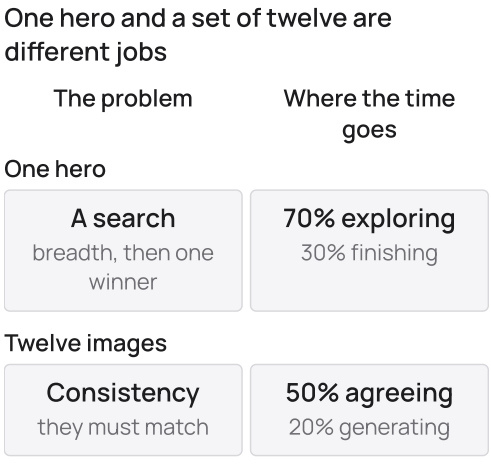
The arithmetic that decides the quote

A rough but honest shape for planning, in generated images rather than hours:

- **One hero**: 60–120 exploration renders, 6–10 full-quality candidates, 20–40 inpainting passes on the winner.
- **Twelve-image set**: 40 renders to find the anchor, then roughly 15–25 renders per frame because you are constraining harder, plus repair on each.

The second is not twelve times the first, but it is comfortably three to four times, and the shape of the work is different: it is more editing and less searching, so it benefits less from a faster GPU and more from a disciplined process.

Figure 2.3



Say which one it is, in writing

The practical outcome of this lesson is a sentence in the spec: "One hero image, plus crops" or "A set of N images sharing one visual system". They price differently, they schedule differently, and a client who asked for one and receives the other is right to be unhappy.

Watch for the request that starts as one and becomes the other. "This is great — can we get five more like it?" is the most common scope expansion in this work, and it is a different job. Five more *like it* means building the system retrospectively from an image that was composed without one, which is harder than building it from the start. The revision lesson later in the course gives you language for that conversation; the thing to do now is notice it is coming.

The honest limit

Consistency across a set has a ceiling that no current method removes. You can hold light, palette, lens and finish. You can hold a character reasonably well with the methods in the consistency block. What you cannot reliably hold

is a *specific* object across many angles — the same chair, with the same scuffs, seen from four sides — because nothing in the model is tracking an object identity between generations.

For those, the honest answers are to photograph it, to model it in Blender, or to design the set so the object is only seen from one angle. Deciding that now, at the brief stage, is what keeps it from becoming a crisis at frame seven.

The order to make a set in

Sequence matters more than people expect, and the productive order is not first-to-last.

1. **The anchor.** The single frame that best embodies the system. Finish it completely, including grade and grain, and get it approved.
2. **The hardest frame.** Whichever one you privately suspect is impossible — the group shot, the specific object, the one with legible text. Do it second, while there is still time to redesign the concept around it.
3. **Everything straightforward.** These go quickly once the system is fixed and the hard problem is solved.
4. **A consistency pass over all twelve together**, on one screen, at the same size. Fix whatever does not belong.

Doing the hard frame second is the part people resist, because it is the least enjoyable. It is also the only order in which discovering a problem is cheap: finding at frame eleven that the concept cannot be delivered means eleven finished frames may need to change.

The final pass is not optional. Images look consistent one at a time and reveal their disagreements the moment they are side by side, which is exactly how the audience will see them.

BEFORE YOU MOVE ON

Why does a twelve-image set need a different method from a single hero image?

A. Because generating twelve images costs roughly twelve times as much compute as one, so the budget has to be allocated far more carefully across the frames.

B. Because each image in a set will be seen for a shorter time than a hero image, so individual frames can be finished to a noticeably lower standard.

C. Because the frames are judged against each other, so the work shifts from searching for one good frame to enforcing constants across all of them.

D. Because a set is more likely to be re-cropped across different placements, which makes crop-safe composition the dominant concern throughout the whole process.

The answer and why: p. 391

Lesson 19 · 7 min

The frames where a phone camera wins

THE ONE THING TO KEEP

Anything that must be exactly itself — a real product, real premises, real staff, food you are selling — is faster, cheaper and safer to photograph than to generate, and recognising those frames early is a professional skill rather than a failure.

The unfashionable answer

Some of the frames on your brief should not be generated at all. Saying so early makes you more useful, not less, and it saves the day that would otherwise be spent discovering it.

The test is simple: **does this image make a claim about a specific real thing?** If yes, generating it is either impossible, dishonest, or more expensive than photographing it.

The categories

The actual product. A generated bottle is a bottle-shaped object with an unreadable label, wrong proportions and a logo that is nearly right. Getting the real one is thirty minutes: a window, a white card as a reflector, a sheet of paper as a background, and a phone on a stack of books. Then generate the *environment* and composite the real product into it, which is a recognised professional workflow and covered later in this course.

Food you are selling. Food photography is regulated in many markets to the extent that the image must represent the actual item. Beyond the law, generated food has a specific failure: it looks delicious and wrong, with impossible textures and garnish that does not correspond to anything. A restaurant that posts generated pictures of its dishes is setting up every customer for disappointment.

Your premises, your staff, your equipment. These are claims about reality. A generated "our team" photograph is a fabrication in a way that a generated abstract background is not, and it is the single fastest way to lose the trust the image was meant to build.

Anything a customer will compare against the real item. Clothing, furniture, hardware. If the generated version is prettier than what arrives, you have manufactured a complaint.

Documentary or journalistic content. No exceptions worth discussing. The later block on rights and disclosure covers this properly.

What generation is genuinely better at

The list is not one-sided, and the frames below are ones where photography would be slower, costlier or impossible:

- **Environments and backgrounds** that do not need to be a specific place.
- **Concepts and abstractions** — growth, connection, an idea in a deck.
- **Things that do not exist** — a product concept, a fantasy scene, an illustration.
- **Sets you cannot afford** — a mountain road at dawn, a 1970s office, a crowded market at blue hour.
- **Texture, pattern, surface** for backgrounds and overlays.
- **Variations for testing** — twelve versions of a banner to see which performs.

The sharpest formulation is this: **generate the world, photograph the claim.**

The phone is enough

The objection is always that there is no camera and no studio. A modern phone in daylight next to a window is a competent product camera, and the gap between a phone photograph and a studio one is mostly lighting, which costs nothing:

1. A window, subject side-on to it, not facing it.
2. A white A4 sheet on the shadow side, angled to bounce light back.
3. A plain background — a sheet of paper, a wall, a piece of card.
4. Phone on something solid. Tap to focus. Turn the flash off.
5. Take thirty and keep two.

Then clean it up. Background removal, colour correction and cut-out are exactly what the image-editing course covers, and all of it is available in GIMP and Photopea without spending anything.

Saying it to a client

"That one we should photograph — it will look better and it will be faster" is a sentence that builds trust, because it demonstrates that you are choosing the tool rather than defending it. The version that damages you is the one where

you generate it anyway, they notice the label is wrong, and you explain afterwards that the model cannot do labels.

The more useful frame for the conversation is that this is normal creative practice. Nobody illustrates a product catalogue; nobody photographs a concept for an annual report. Choosing per frame is the job, and it has always been the job.

The middle category

Between "photograph it" and "generate it" sits a third answer that handles a surprising share of briefs: **photograph the claim, generate everything around it**.

The bottle is real and shot against a white sheet. The mountain road it is standing on does not exist. The composite is honest because the product is genuinely what it is, and it is affordable because nobody drove to a mountain.

The same logic applies widely. A real headshot placed into a generated environment. A real building photographed on a dull day with a generated sky, which retouchers have done for thirty years without anyone calling it deception. A real dish on a generated table.

Where the line sits is a judgement, and it is worth stating the principle rather than a rule: **the parts of the image a viewer would rely on should be real, and the parts that are stage dressing need not be.** A sky is stage dressing. A dish, in an advertisement for that dish, is not.

When you are unsure, the test that has never failed anybody is to imagine explaining the composite to the person it depicts, or to the customer who buys on the strength of it. If that explanation is comfortable, the composite is fine. If it is not, you already know.

BEFORE YOU MOVE ON

What distinguishes the frames that should be photographed from the ones that should be generated?

A. Whether the subject contains fine detail such as text or a logo, since those are the specific elements that current generative models reproduce least reliably in practice.

B. Whether the image asserts something about a specific real thing that a viewer could check against reality.

C. Whether photographing it would be cheaper than the compute and time that generating an acceptable version of the same frame would consume.

D. Whether the final image will be seen at a large enough size for generated artefacts and inconsistencies to become visible to an ordinary viewer.

The answer and why: p. 392

Lesson 20 · 7 min

The one page that ends the revision argument

THE ONE THING TO KEEP

A signed one-page spec that names dimensions, ratio, forbidden content, the reference pack and what will be photographed rather than generated converts later disputes from matters of taste into matters of record.

What the document is for

Not bureaucracy. The spec exists so that the sentence "this isn't what we wanted" has somewhere to be resolved. Without it, every review is a negotiation about remembered intentions. With it, a review is a comparison between an image and a page you both agreed to.

It fits on one side. If it runs longer, it has become a proposal and something else has gone wrong.

The template

```
# Image spec — [client] — [date]

## Deliverables
- Hero: 1200 × 1500 px, JPEG, sRGB
- Crops from the same frame: 1080 × 1080, 1080 × 1920
- Print version: A4 at 300 dpi, CMYK, 3 mm bleed

## Generation plan
- Native render ratio: 896 × 1152 (SDXL bucket), upscaled and
cropped
- Crop-safe core: central square; headline sits in the upper
third

## The picture
One sentence describing the image. Not a prompt — a sentence a
person understands.

## Reference pack
1. light — ref-01.jpg — the hard side light, not the styling
2. palette — ref-02.jpg — the muted greens
3. composition — ref-03.jpg — subject low and left, lots of sky
4. product — ref-04.jpg — our actual bottle, photographed 14 Mar
5. texture — ref-05.jpg — matte, fine grain, not glossy
6. anti-reference — ref-06.jpg — too clean, too corporate

## Constraints
- Brand red #E4002B, exact, on the label only
- Headline zone: upper third must be quiet, contrast ratio at
least 4.5:1

## Must not appear
- Legible brand names or logos of any kind
- Text other than type we set ourselves
- Hands

## Photographed, not generated
- The bottle (composited into a generated environment)

## Rounds
Two rounds of revisions included. Structural changes — different
composition,
different concept — are a new round.
```

```
## Disclosure
Client is informed the environment is AI-generated. Delivery
includes a note
for their records.
```

The four lines that do the work

Most of that template is useful. Four lines are load-bearing.

"Photographed, not generated" sets the expectation that some elements are real before anyone has an opinion about the picture. It is far easier to state on day one than to explain on day six.

"Must not appear" is the list a person checks against the finished file. It is the difference between an embarrassment caught in QA and one caught by the client's legal team.

The rounds clause distinguishes a tweak from a rebuild. "Can we make it warmer" is a tweak. "Can she be facing the other way" is a regeneration that loses everything downstream of the original frame, and the spec is where that gets priced.

The disclosure line means nobody is surprised. The rights block later covers what you must say and where; the point here is that it is agreed at the start rather than discovered at the end.

Getting it signed without ceremony

You do not need a legal process. An email saying "here's the spec — confirm and I'll start" and a reply saying "confirmed" is enough for the purpose, which is a shared record rather than an enforceable instrument.

What matters is the timing: before generating. A spec written after the first review is a defence, and everyone can tell.

Its real limit

A spec cannot make a client know what they want. Some genuinely do not until they see something, and insisting on precision before any image exists can stall a project that a quick sketch would unblock.

So use it proportionately. For a small job, three lines in an email — dimensions, what must not appear, how many rounds — carries most of the value. For anything with print, legal exposure or more than a few images, write the page. And if a client cannot answer the questions yet, generate three cheap, deliberately rough concepts to make the conversation concrete, then write the spec from whichever one they point at. The spec is a tool for removing ambiguity, not a gate that ambiguity has to pass first.

Keep it alive

A spec written once and never touched becomes fiction by the third day, because briefs move. The version that stays useful is edited as decisions are made, with a short note of what changed and when.

```
## Changes
- 14 Mar: headline moved to lower third — upper third now free
- 16 Mar: bottle to be photographed, not generated (label
unreadable in tests)
- 18 Mar: second size added, 1080 × 1920
```

Three lines like that resolve more disputes than the entire rest of the document, because the arguments that actually happen are almost never about the original brief. They are about a change somebody remembers differently.

A last, unglamorous point: keep the spec beside the work, in the project folder, not in a chat thread. Chat histories are searched badly and scroll away. A file called `brief.md` sitting next to the images is still there in a year, when someone asks why the composition has all that empty space at the top and nobody remembers that a headline was supposed to go there.

BEFORE YOU MOVE ON

What does an image spec actually accomplish at review time?

A. It transfers responsibility for any content problem in the delivered file from the person who made the image to the client who approved the specification document.

B. It documents the exact technical parameters in enough detail that the approved image could be regenerated later from the specification alone.

C. It turns a disagreement about taste into a comparison against a record both sides agreed to beforehand.

D. It allows the number of revision rounds to be enforced, which is the main commercial protection the document provides on a fixed-price job.

The answer and why: p. 392

Lesson 21 · 10 min

What you can sell, and what you must disclose

THE ONE THING TO KEEP

Being licensed to use a model, owning its output, and having to disclose that output is generated are three separate questions, and only the ownership one is genuinely unsettled.

Four questions, routinely mashed into one

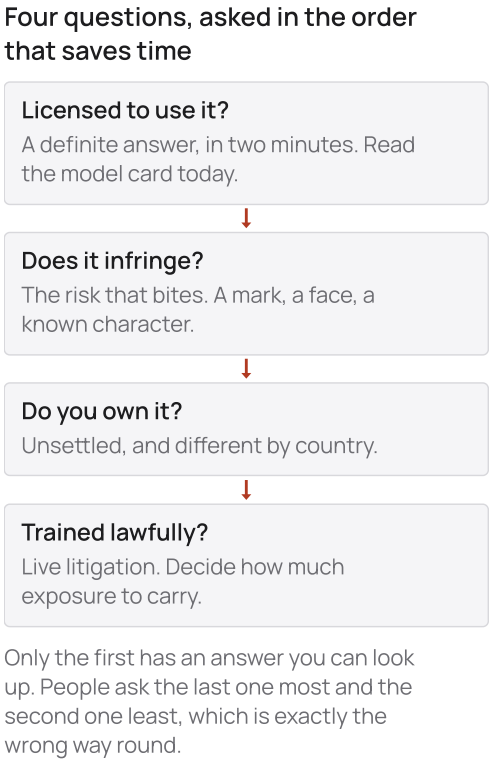
People ask "can I use AI images commercially" as though it were a single question. It is four, they have different answers, and only one of them is genuinely unsettled.

1. Was the model trained lawfully?
2. Are you licensed to use this model?

3. Do you own the output?
4. Does the output infringe somebody else?

None of what follows is legal advice. It is the shape of the problem. A lawyer in your own country is the person who answers it for your case.

Figure 2.4



One: was the model trained lawfully

Live litigation, several countries, no settled answer. Getty sued Stability in the US and the UK. Artists have brought class actions in the US. Publishers and image libraries have brought proceedings in Europe. Some cases have narrowed, some have settled quietly, none has produced a clean rule.

You will not resolve this and you do not have to. What you can do is decide how much exposure to it you want to carry, which leads to the tools that offer indemnity, below.

Two: are you licensed to use the model

This one has a definite answer and you can look it up in two minutes. It is lesson one's licence trap.

- Open weights carry a licence. Apache 2.0 releases such as FLUX.1 [schnell] and Qwen-Image permit commercial use. FLUX.1 [dev] does not. Stability's recent community licence is free below a revenue threshold and paid above it.
- Hosted tools carry terms of service, and free tiers frequently restrict commercial use where paid tiers do not.

These positions were accurate in 2025 and they change. Read the model card or terms page on the day the job starts.

Three: do you own the output

Genuinely unsettled, and different by country. The short version:

- **The United States** requires human authorship. Prompts alone do not make you an author. Your own expressive contributions — a drawing you fed in, your edits, your arrangement — can be protected.
- **The United Kingdom** and a group of jurisdictions with similar provisions, including India, deem the author of a computer-generated work to be the person who made the arrangements for its creation.
- **China's** Beijing Internet Court reached the opposite conclusion to the US on similar facts, finding protection where the user's choices showed personal intellectual investment.
- **The European Union** has no harmonised answer and its member states are working it through.

The same image can be unprotectable in one country and protected in another. Making things with AI sets out the cases in detail.

The practical consequences: a prompt is not property, "you own your outputs" in a tool's terms is a promise not to claim it rather than a right the company can create, and for a logo you should route around the problem entirely —

trademark protects a mark through use and registration however it was drawn.

Four: does it infringe somebody else

This is the risk that actually bites, and it is the one people ask about least. A recognisable cartoon character, a real person's face, a trade dress, a style so specific it names a living artist. That exposure exists whether a human or a model drew it.

Indemnification is what Adobe, Getty and Shutterstock offer on business plans: if a claim arrives about the training data behind their generator, they will stand behind it. Read the exclusions. It covers what the model learned from. It does not cover you prompting a famous mouse.

Labelling: moving fast, in one direction

- **China** has required both a visible label and a metadata label on synthetic content since September 2025, with a duty on platforms to check.
- **India** amended its IT Rules for synthetically generated information and prescribed *visibility* — a label covering a defined portion of the frame and the opening portion of audio, plus a platform duty to verify user declarations.
- **The European Union's** AI Act requires machine-readable marking of generated output and disclosure of deepfakes; the timetable has been amended more than once, so check the current position.
- **The United States** has no general federal labelling law, a 2025 statute on non-consensual intimate imagery, and a patchwork of state election rules.

The technical layer — C2PA Content Credentials, and invisible watermarks such as Google's SynthID — is real and fragile. A screenshot destroys it. Many re-uploads strip it. It is worth attaching and it is not a substitute for a label a person can see.

The line for telling a client

Tell them, in writing, whenever a reasonable viewer could take the image as a record of something real: a person, a place, a product's actual appearance, a customer, an event. Tell them also when they will license the asset onward, because the ownership question becomes theirs and they need to know what they are selling.

One line in the delivery note does it:

> *Assets 3, 5 and 7 are AI-generated. They do not depict real people or real events. Generated with [tool] under [plan]; source files and settings supplied.*

That sentence has never lost anybody a job. Discovering it later has.

Today: write that line into your own delivery template, before you need it.

BEFORE YOU MOVE ON

An agency in India delivers a generated photograph of a smiling "customer" for a client's testimonial page, with no label, arguing that the tool's terms grant them ownership of outputs so there is nothing to disclose. What is wrong with that reasoning?

A. Nothing, provided the tool's terms genuinely assign output ownership to the paying user.

B. The file only needs a C2PA Content Credential attached to satisfy the requirement.

C. Ownership and disclosure are separate questions; a synthetic image presented as a real customer is precisely what labelling rules address, India's IT Rules prescribe a visible label, and the client should have been told in writing that the asset is generated.

D. Synthetic-media rules cover video deepfakes of identifiable individuals, so a still image of an invented person falls outside them.

The answer and why: p. 393

CASE STUDY · MODULE 2

The coaching centre that wanted a campus it did not have

An illustrative case, built from documented patterns.

A coaching institute in Kota commissioning a prospectus and an admissions campaign six weeks before the enrolment season.

The brief arrived as one line in a WhatsApp message: something premium for admissions, like the big chains. The institute had two floors above a bank, one hundred and ninety students, and a genuinely good physics faculty. The director's reference was a competitor's brochure showing a landscaped campus, a library with a double-height ceiling, and eleven smiling students on a lawn.

Anusha, the designer, did the thing the brief lesson asks for before opening any generator: she asked where each image would appear, and she asked the fourth question — does anything real have to be in it.

The answers produced a list of eleven images. Four were straightforward: an abstract cover, two section dividers, a background for a results panel. Four were environments — a study scene, a library feel, a revision-at-night mood, a corridor — where nothing depicted a specific real place. And three were the problem: a photograph captioned *our campus*, a group described as *our students*, and a portrait row labelled *our faculty*.

The director wanted all eleven generated. His reasoning was not dishonest in his own mind. The building looks tired, he said, and students judge on photographs, and every institute in this town shows a campus that is nicer than it is. Photographing the actual premises would produce something that looked worse than the truth, because the corridor lighting is bad and the classrooms are full.

Anusha's position was that the four environments were fine to generate, the four abstracts obviously so, and the three labelled ones were not. A generated building captioned *our campus* is not a stylistic choice; it is a claim about a

place that does not exist, made to sixteen-year-olds and their parents deciding where to spend a year's savings. The same for a room of generated students and a row of generated faculty.

The cost of saying so was real. The institute had been recommended to her, the fee was the largest single job in her quarter, and the director had made it clear that another designer would do it if she would not. She also knew that he was right about one thing: the competitor's brochure almost certainly contained a generated lawn.

The cost of not saying so was also real, and it was not only ethical. Parents visit. A parent who has driven from Bundi on the strength of a photograph of a library, and then sees the actual first floor, is a parent who tells other parents. The image that was supposed to build trust is the one that removes it, and it does so at the worst possible moment, in person, with the director in the room.

So she wrote the spec before arguing. One page. Deliverables with pixel sizes. Native render ratios. The forbidden list — no legible brand names, no text other than type we set, no recognisable real landmarks. And then two lines that were the whole negotiation:

Photographed, not generated: the premises, the faculty, and any image captioned as depicting this institute.

Generated: environments, textures and abstract backgrounds that do not depict a specific place or person.

She took the spec to the meeting rather than an opinion, and she brought an alternative rather than a refusal: one afternoon with a phone, a window, and a white card, photographing four faculty members properly and the two best corners of the building at the right hour. Then the environments generated around them, which is a composite and not a fabrication.

YOUR ANSWERS

1. The director argued that his competitors were already doing this. Why is that not a reason to do it, in terms the brief lesson would use?

2. Anusha brought a written spec instead of an opinion. What did the document actually do in that meeting?

3. Which of the eleven images were legitimate to generate, and what made them different?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 112.

CASE STUDY · MODULE 2

How it played out

The director agreed to the two lines, mostly because the alternative was concrete rather than because he was persuaded by the principle. The photography took four hours on a Sunday and cost nothing. Three of the four faculty portraits were usable; the fourth needed reshooting and was reshot. The generated environments — a night revision mood, a corridor, two study scenes — were kept and clearly read as illustration rather than record, which was the point. The prospectus shipped a week late because of the reshoot. Six months later the institute's counsellor mentioned, unprompted, that several parents had said the faculty photographs were what made the place feel real; the competitor's brochure, she said, looked like every other brochure. Anusha now sends the two-line version of the spec with every quotation, before the fee is agreed, because saying it on day one is expertise and saying it on day six is an excuse.

The questions, discussed

1. The director argued that his competitors were already doing this. Why is that not a reason to do it, in terms the brief lesson would use?

Because the test is not what others do but whether the image makes a claim about a specific real thing. A landscaped lawn on a competitor's brochure is the same failure, not a permission. The parent who visits compares the photograph against the building, and the image that was meant to build trust is the one that destroys it — at the point of decision, in person.

2. Anusha brought a written spec instead of an opinion. What did the document actually do in that meeting?

It converted a disagreement about taste into a record. With a spec, the review is a comparison between an image and a page both parties agreed to; without one, every review is a negotiation about remembered intentions. The two load-bearing lines — what is photographed and what must not appear — set the expectation before anybody had an opinion about a picture.

3. Which of the eleven images were legitimate to generate, and what made them different?

The four abstracts and the four environments, because none of them depicts a specific place, person or product a viewer would rely on. They are stage dressing. The three labelled images — campus, students, faculty — were claims about reality, and a generated claim is a fabrication however good the picture is.

MODULE 2 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. 387. If two go wrong, re-read the module before the exam.

1 Why does an SDXL render at 1024 × 2048 often contain two torsos?

A. The unusual ratio confuses the text encoder's positional weighting.
B. The extra pixels exceed the card's memory budget, so the image is generated in two halves and joined.
C. The canvas is far outside the trained buckets, so two regions each resolve as a centre.
D. The VAE cannot decode a non-square latent and tiles it instead.

2 You need 1200 × 630 and the nearest SDXL bucket is 1344 × 768. What is the recommended route?

A. Generate at the bucket and stretch it vertically to the delivery height, which is imperceptible at this scale.
B. Generate at the bucket and outpaint the sides rather than cropping top and bottom.
C. Generate at 2048 × 1024 so there is plenty of detail to crop from.
D. Generate at 1200 × 630 directly, since it is comfortably under a megapixel.

3 What does designing a frame for crop safety actually cost you?

A. A more centred, more conservative frame than one composed for a single format.
B. Nothing, since it is a free technique that improves every composition.
C. The ability to place type, because the quiet zone has to stay empty.
D. Resolution, because the file you deliver is only the surviving part of a larger render.

4 Why does "no logos" in the negative prompt not protect a client from a trademarked shape appearing on background signage?

A. Logos are introduced by the upscaler rather than by the base model.
B. The negative prompt is capped at 77 tokens, so a long list is truncated.
C. Negative prompts are read only by CLIP-based models and ignored by the FLUX family.
D. The negative prompt biases the outcome; it does not forbid anything.

5 A brand red is specified as #E4002B. What is the reliable way to hit it?

A. Use a negative prompt listing every neighbouring colour you do not want.
B. Put the hex value at the front of the prompt, where earlier tokens are weighted more heavily.
C. Generate the element neutral and set the exact value in an editor afterwards.
D. Raise the guidance scale until the colour saturates toward the specified value.

6 A client approves one hero image and then asks for five more like it. Why is that a different job rather than a small extension?

A. Five more images cost five times the credits the first one did.
B. The visual system has to be built backwards from a frame composed without one.
C. The seed for the approved frame will not reproduce in a later session.
D. Consistency requires a character LoRA, and the first image was made without one.

3

MODULE 3

Getting to a frame worth keeping

Run a structured exploration that reaches a usable frame inside a stated budget — a prompt assembled in named slots, a low-cost contact sheet judged on a grid, seed-fixed single-variable comparisons — and apply an explicit test for whether a frame is worth taking into repair or should be abandoned.

22 Build the prompt in slots, not as a sentence you keep rewriting 118
23 The camera vocabulary that works, and the words that do nothing 123
24 Name the light or accept the model's average 128
25 Describing a look without naming a living artist 132
26 Sixteen cheap frames beat four expensive ones 136
27 Fix the seed or learn nothing 140
28 Change one thing, or you have learned nothing 145
29 Recognising the twenty minutes that will never converge 149
30 Interrogating a picture for vocabulary 153
31 What "good enough to take forward" actually means 157
Case study: Forty minutes on seven boats 163
Module 3 test 168

Lesson 22 · 8 min

Build the prompt in slots, not as a sentence you keep rewriting

THE ONE THING TO KEEP

Writing a prompt as seven named slots makes it debuggable, because when the image is wrong you can change one slot and know which part of the description caused the change.

Why prompts get worse as you edit them

Watch somebody work on a prompt for twenty minutes and you see the same pattern. They add a word, generate, do not like it, add three more, generate, delete something, add "highly detailed", generate. After twenty minutes the prompt is ninety words long, nobody can say what any of it is doing, and the images are not better than attempt four.

The problem is that the prompt has no structure, so there is nothing to change *one of*. The fix is to give it slots.

Seven slots

Write the prompt as an ordered list of named parts. Keep the names in your notes, even if the model only ever sees the joined text.

1. **Medium** — photograph, oil painting, technical illustration, 3D render
2. **Subject** — who or what, with the two or three attributes that matter
3. **Action or state** — what they are doing, or explicitly at rest
4. **Setting** — where, and how much of it is visible
5. **Light** — direction, quality, time of day, colour
6. **Camera** — framing, focal length, height, depth of field
7. **Finish** — grain, film stock, process, colour treatment

A worked example:

```
medium:  documentary photograph
subject: an older woman in a green cotton sari, silver bangles
action:  sorting dried chillies on a low table, hands in motion
setting: an open courtyard, whitewashed wall behind, few props
light:   hard late-afternoon sun from the left, deep shadows
camera:  waist-level, 35mm, full body with headroom, slight wide
angle
finish:  colour negative film, fine grain, slightly cool shadows
```

Joined with commas, that is a prompt. Kept as a list, it is a specification you can debug.

Figure 3.1

Seven named slots, so there is something to change one of

Medium
Photograph, oil painting, technical illustration, 3D render

Subject
Who or what, with the two or three attributes that matter

Action or state
What they are doing, or explicitly at rest

Setting
Where, and how much of it is visible

Light
Direction, quality, time of day, colour

Camera
Framing, focal length, height, depth of field

Finish
Grain, film stock, process, colour treatment

Light is the slot missing most often, and the one whose absence makes an image look generated. Slots five to seven lift straight out into the next job as a house look.

What the slots buy you

Coverage. Most weak prompts are missing a slot entirely, and light is missing most often. Left unspecified, the model gives you its average, which on photographic checkpoints means the soft, even, everywhere-at-once light that makes generated pictures recognisable as generated.

Debugging. The image is too tight. That is the camera slot, not the subject. The colours are wrong. That is finish or light. Without slots, people respond to a framing problem by changing the subject description, and then cannot explain why it got worse.

Reuse. Slots 5, 6 and 7 are the visual system from your brief. Fix them across twelve images and you have the consistency the set block will ask for, at no extra cost.

Order, and how much it matters

It matters, but less than it used to, and it matters differently per architecture.

CLIP-based models — SD 1.5, SDXL — read the prompt in 75-token chunks and weight earlier tokens somewhat more heavily. Put the subject early there.

T5-based models — the FLUX family and several newer systems — handle longer natural-language descriptions and are far less position-sensitive. With those, full sentences outperform comma-separated tags, and burying an important detail at the end is much less punishing.

The practical rule that survives both: **put the thing the image is about in the first fifteen words, and do not rely on anything after word sixty.** Which brings us to the real ceiling.

The limit nobody tells you

Adding more words stops working, and the mechanism is worth knowing. CLIP text encoders have a 77-token window. Interfaces get around it by splitting longer prompts into chunks, encoding each separately, and concatenating the results. There is no attention across the chunk boundary, so a relationship you describe in words 80 to 90 cannot bind to a subject named in word 5.

That is why enormous prompts produce images containing all the right *things* in none of the right relationships. The model is not ignoring you; the architecture has no path for that particular instruction to reach that particular subject.

So when a prompt passes about sixty words and is still not working, stop adding. Either move the requirement to a different mechanism — a reference image, a control, an inpainting pass — or accept that this part gets fixed in editing. The next block is entirely about that first option, and it is where prompts stop being the main tool.

Keep the slots as a file

The practical form of this is a small text file per project rather than a prompt box you type into.

```
# courtyard hero — slots
MEDIUM   documentary photograph
SUBJECT  older woman, green cotton sari, silver bangles
ACTION   sorting dried chillies on a low table, hands in motion
SETTING  open courtyard, whitewashed wall, few props
LIGHT    hard late-afternoon sun from the left, deep shadows
CAMERA   waist level, 35mm, full body with headroom
FINISH   colour negative film, fine grain, cool shadows
```

Three things follow from having it on disk. You can diff two versions and see exactly what changed between Tuesday and Thursday. You can hand it to somebody else and they can work from it. And slots 5 to 7 lift straight out into the next job as a house look, which is where a personal style starts to come from.

It also stops the slow creep. When the prompt lives in a box you keep editing, it only ever grows. When it lives in named fields, adding a word means deciding which field it belongs to, and about a third of the time you realise it belongs to none of them and should not be there.

BEFORE YOU MOVE ON

A ninety-word prompt describes a woman in a blue coat and a man in a red scarf, and the model keeps swapping the colours. Adding emphasis does not fix it. What is the mechanism?

A. Colour words are weakly represented in the text encoder, so they bind to the scene as a whole rather than to individual subjects mentioned in the prompt.

B. Attribute binding weakens across the 77-token chunk boundary, so a long prompt loses the link between a subject and its attributes.

C. The guidance scale is too low for the model to enforce two separate attribute assignments within one image at the same time.

D. The two subjects are too similar in the training distribution, so the model treats them as one concept with two interchangeable sets of attributes.

The answer and why: p. 395

Lesson 23 · 8 min

The camera vocabulary that works, and the words that do nothing

THE ONE THING TO KEEP

Focal length, aperture, camera height and shot size measurably change composition because they appeared in the captions of millions of training photographs, while quality words like "8k" and "masterpiece" mostly do nothing on modern models.

Why photographic words work

A model learned from images with captions, and a very large number of those captions came from photography sites where the metadata was attached: focal length, aperture, shot type. So the model has seen tens of thousands of

examples of what "85mm f/1.8" looks like — compressed perspective, shallow depth of field, a particular subject distance — and the phrase reliably produces that.

This makes photographic vocabulary the most efficient lever available. One term replaces a paragraph of description.

The terms that reliably change the image

Focal length changes perspective, not just zoom:

- `24mm` or `wide angle` — broad view, exaggerated depth, distortion near edges. Good for environments and for leaving headline room.
- `35mm` — the documentary standard. Natural, slightly wide, subject in context.
- `50mm` — close to human vision. Neutral.
- `85mm` — portrait compression. Flattering, background falls away, tighter frame.
- `135mm` or `200mm` — heavy compression, background collapses toward the subject.
- `macro` — very close, extremely shallow focus.

Aperture controls the background:

- `f/1.4`, `f/1.8` — background dissolved, one plane sharp.
- `f/5.6` — subject and near context sharp.
- `f/16` — everything sharp, deep landscape look.

Camera height and angle, which is the most under-used control of all:

- `low angle`, `from below` — the subject gains stature.
- `eye level` — neutral, the default.
- `high angle`, `from above` — the subject diminishes.
- `overhead`, `top-down` — flat-lay, for objects and food.
- `waist-level` — the documentary look, and the one that stops portraits feeling posed.

Shot size, borrowed from film and instantly understood:

- `extreme wide shot`, `wide shot`, `medium shot`, `close-up`, `extreme close-up`.

The words that do nothing

On SD 1.5, quality tags had measurable effect because community fine-tunes had been trained with them in the captions. On modern models they are mostly inert or actively harmful:

- `8k`, `4k`, `ultra HD` — resolution is set by the image size, not by words. On some checkpoints they nudge toward an over-sharpened stock-photo look.
- `masterpiece`, `best quality`, `award winning` — largely inert on newer models; on some anime-lineage checkpoints they still do something, which is why the habit persists.
- `highly detailed`, `intricate` — tends to fill the frame with ornament, which is usually not what you meant.
- `trending on ArtStation` — a phrase with a specific historical effect on an old model that people still copy into models where it means nothing.

Test them rather than trusting either me or the forum: fix the seed, generate with and without, and look. On most current checkpoints you will see almost no difference, and the sixty characters are better spent on light.

The failure this vocabulary causes

There is a real cost to leaning on it, and it is worth naming because it is the most common way competent prompts produce boring pictures.

Photographic terms pull toward the *centre* of what that term looked like in the training data. Ask for `85mm portrait, f/1.8, soft light` and you will get an extremely competent, extremely average portrait — the statistical middle of a million flattering headshots. The vocabulary is precise, which means it is precisely conventional.

The way out is to use the technical slot for control and put the interest somewhere else: an unusual subject, an odd moment, a specific place, an awkward gesture. Or to deliberately combine terms that rarely co-occur — a macro lens on a landscape subject, a low angle on a domestic scene — which pushes the model off the well-worn path.

One more practical note: these terms describe an appearance, not an optical simulation. There is no camera. `f/1.4` produces the *look* of shallow depth of field, applied by a model that has no depth information, which is why the blur sometimes falls in the wrong place — a hand at the same distance as the face going soft, or a background object mysteriously sharp. When that happens, the fix is a real depth-of-field pass in an editor, not a different aperture number.

Film and video terms, which also work

The same logic extends beyond stills, because film stills and production photographs were captioned too.

- `anamorphic` — wide frame, horizontal flares, oval highlights in the background blur.
- `handheld` — slight imperfection in framing, a documentary immediacy.
- `dutch angle` — tilted horizon. Use sparingly; it is a strong signal.
- `shot on 16mm` — grainier, contrastier, a particular period feel.
- `establishing shot` — pulls the camera back and includes the environment.

And two terms from the studio that do a great deal of work for product and still life: `seamless backdrop` gives you the infinite white or grey sweep behind a product, and `tabletop, hero angle` gives the slightly-above three-quarter view that almost every product photograph in the world uses.

A caution about stacking

Every one of these terms narrows the distribution, and four or five of them together narrow it hard. `85mm, f/1.4, golden hour, shallow depth of field, bokeh` is five constraints that all point at the same well-trodden place,

and the result will be the most predictable image the model can make.

Use two or three. If you find yourself adding a fifth camera term, the problem is almost certainly in a different slot.

BEFORE YOU MOVE ON

Why do focal-length terms like "35mm" and "85mm" change an image more reliably than "8k, masterpiece"?

A. Because numeric terms are tokenised more precisely than adjectives, which gives the text encoder a sharper signal to condition the denoising process on.

B. Because focal length physically determines perspective, and the model has learned an internal optical simulation of how lenses project a scene.

C. Because focal lengths appeared in enormous numbers of training captions attached to a consistent look, whereas quality tags did not.

D. Because quality tags are usually placed at the end of a prompt, where the encoder weights them less heavily than terms appearing near the beginning.

The answer and why: p. 396

Lesson 24 · 8 min

Name the light or accept the model's average

THE ONE THING TO KEEP

Light is the single highest-leverage slot in a prompt, and leaving it unspecified is the main reason generated images look generically generated.

The tell you can fix in one line

Put two generated images side by side: one with no lighting described, one with. The first has soft, even, directionless illumination with weak shadows, faces lit from everywhere, and a slight glow. It is the average of millions of photographs, and averaging light is what produces that look.

Photographers and cinematographers spend most of their working lives on light because it carries almost everything: mood, depth, time, weather, whether a scene feels safe or not. The same is true here, and it costs one slot in your prompt.

Vocabulary that lands

Direction, which is the most important and most neglected:

- `side light`, `lit from the left` — texture, form, drama
- `backlit`, `rim light`, `contre-jour` — outline, separation, haze
- `top light` — harsh, institutional, unflattering
- `underlit` — unsettling; almost never accidental
- `front light` — flat, documentary, the passport-photo look

Quality, meaning hard or soft:

- `hard sunlight`, `direct sun`, `harsh shadows` — sharp-edged shadows
- `soft light`, `overcast`, `diffused`, `large softbox` — gradual shadows
- `dappled light through leaves` — a specific, very effective texture

Time and weather, which bundle direction, quality and colour into one phrase:

- `golden hour`, `blue hour`, `midday sun`, `overcast afternoon`, `pre-dawn`
- `foggy`, `after rain`, `humid haze`

Source, which sets colour and mood:

- `window light`, `candlelight`, `fluorescent office lighting`, `sodium street lamp`, `screen glow`, `neon`, `single bare bulb`

Ratio, borrowed from the studio:

- `high contrast, deep shadows`, `low key`, `high key`, `chiaroscuro`

Three combinations worth memorising

For a portrait that does not look generated: `window light from the left, soft, deep shadows on the right side of the face, overcast afternoon`. Directional but soft, and the shadow side does the work.

For a product that looks like it was shot properly: `single large softbox from above and slightly behind, black card on the left for a dark edge, dark background`. That dark edge along one side is what separates a real product shot from a floating object.

For a documentary feel: `hard late-afternoon sun, long shadows, lens flare, shot into the light`. Imperfect on purpose.

Why this fixes more than lighting

Describing light also fixes composition and mood as a side effect, because light and composition are not separable. "Backlit in a doorway" decides where the subject is, what the background does, and how the eye moves through the frame. That is three slots' worth of control from one phrase.

It also solves the most common complaint about generated images — that they look plastic. Plastic is not a texture problem. It is a light problem: no dominant source, no real shadow, no falloff. Give the scene one strong source

and a direction, and the plastic mostly goes.

The limitation to expect

The model does not compute light. There is no source, no ray, no surface. It reproduces the *appearance* of a lighting setup, which means the light is right where the training data was dense and wrong where it was not.

Concretely: shadows that do not agree on a direction, a reflection that does not correspond to anything in the room, a mirror showing a scene that is not the scene, a light source visible in frame that casts nothing. These get worse with more objects and worse again with reflective or transparent materials — glass, water, chrome, polished floors.

You catch them by looking, deliberately, at a checklist: every shadow should agree on a direction; every reflective surface should reflect something plausible; anything glowing in frame should be illuminating what is next to it. Two of the three will usually be fine and one will not, and that one is an inpainting job rather than a prompting one. Nobody's eye catches these automatically at first. It is a habit you build in about a fortnight, and after that you cannot stop seeing them.

Learning light without a studio

The fastest way to build this vocabulary costs nothing and does not involve a generator at all.

Take a single object — an onion, a mug, a shoe — and photograph it eight times in one afternoon with a phone. By a window at 10am. By the same window at 4pm. With a white sheet of paper bouncing light back on the shadow side. With a dark cloth there instead. Under the overhead room light. With a phone torch held to one side, then above, then below.

Eight photographs of the same object, and the difference between them is enormous. Look at where the shadow falls, how sharp its edge is, how much of the form you can read, and what each one makes the object feel like.

This is a real afternoon and it is worth more than any amount of reading, because it attaches the words to something you have seen happen. Afterwards, `hard light from the left` is not a phrase you copied from a prompt list; it is a thing you have made and can predict. That is the difference between describing light and choosing it.

BEFORE YOU MOVE ON

Generated portraits look plastic despite a detailed subject description. What is the most likely cause?

A. The checkpoint is a community merge whose averaging has pushed its output toward an over-smoothed skin rendering that no prompt can fully counteract.

B. No light is specified, so the model produces its soft directionless average, which reads as plastic.

C. The prompt lacks skin-texture vocabulary such as pores and fine lines, which the model needs in order to render a surface as anything other than uniformly smooth.

D. The guidance scale is set too high, which smooths fine detail as the sampler is pushed harder toward the prompt on every step.

The answer and why: p. 396

Lesson 25 · 8 min

Describing a look without naming a living artist

THE ONE THING TO KEEP

A look is more reliably reproduced by naming its process, era, materials and printing than by naming an artist, and the process description also avoids the ethical and commercial problem of trading on a living person's name.

Two reasons to stop typing artist names

The first is practical. "In the style of [artist]" is a blunt instrument. It pulls in subject matter, palette, composition and period all at once, so you get their *pictures* rather than their *technique*, and you cannot adjust one part of it. Ask for a technique instead and every component stays under your control.

The second is that using a living artist's name to make commercial work that competes with them is, at minimum, a bad thing to do. The legal position on style is genuinely unsettled and differs by country — style as such has generally not been protectable by copyright, while passing off, moral rights and unfair competition doctrines vary considerably, and cases are live in several jurisdictions. That is a real disagreement, not a settled rule, and this course is not the place to resolve it. What can be said without controversy: naming a living working artist in a prompt for paid work is a decision you would struggle to defend to them, and there is a better method anyway.

Long-dead artists and historical movements are a different matter and are ordinary art-historical vocabulary.

Describe the process instead

A look is made of specific physical choices. Name those.

Photographic process: `wet plate collodion`, `daguerreotype`, `cyanotype`, `Polaroid SX-70`, `35mm colour negative`, `Kodachrome`, `cross-processed E-6`, `push-processed Tri-X`, `medium format transparency`.

Printing and reproduction: `letterpress`, `risograph, two colours, slight misregistration`, `newsprint halftone`, `screen print, flat inks`, `offset with visible rosette`, `photocopied and rescanned`.

Illustration materials: `gouache on rough paper`, `ink wash`, `coloured pencil on toned ground`, `linocut`, `scratchboard`, `airbrush on board`.

Era and context, which carry a whole visual world: `1970s editorial photograph`, `1950s advertising illustration`, `Soviet poster`, `1990s point-and-shoot flash snapshot`, `early web graphic`.

Rendering approach for digital looks: `flat vector, limited palette`, `isometric, no perspective`, `clay render, ambient occlusion only`, `technical cutaway diagram`.

Why this reproduces better

Process vocabulary describes *how the image was made*, which is exactly what produced the consistent appearance in the training data. Every risograph print in that data shares the same slight misregistration, the same ink-on-uncoated-paper texture, the same limited palette. So the phrase has a tight, consistent meaning.

An artist's name has a diffuse one: their early work, their late work, their most-reproduced pieces, other people imitating them, and whatever the captions happened to say. That is why artist-name prompts are unstable across seeds while process prompts are steady.

Process terms also **stack cleanly**. `risograph, two colours` plus `1970s public information poster` plus `flat vector` combine into something coherent, because they are describing compatible aspects of a single object. Two artist names average into mush.

Building a style you own

The most valuable version of this is a style description that is yours — a fixed combination of process, palette, light and finish that you reuse. Three or four such recipes, each three lines long, tested and written down, is a more useful asset than a folder of prompts, because it makes your work recognisable across jobs without depending on anyone else's name.

```
house look A
  medium/finish: 35mm colour negative, fine grain, slight
halation
  palette:       muted greens and warm neutrals, no pure black
  light:         window light, one direction, deep soft shadows
  camera:        35mm, waist level, generous headroom
```

Where this still falls short

Process vocabulary controls surface and palette well. It controls **drawing** badly. What makes an illustrator distinctive is usually how they simplify a form — which lines they keep, how they abstract a hand, what they exaggerate — and no amount of material description reaches that.

So if you genuinely need a particular way of drawing, the honest answers are to train a LoRA on work you have the right to use, which the consistency block covers, or to commission an illustrator. A prompt will not get you there, and this is one of the places where the gap between generated and made work is still wide and visible to anyone who looks at illustration for a living.

What to do when a client names an artist

They will. "Can we have it like [well-known living illustrator]?" is a normal sentence in a briefing, and it is usually not a request to copy anybody — it is the only language they have for a look.

The useful response is to translate rather than refuse. Find two or three examples of the work, and describe what is actually producing the effect: the palette is four flat colours with no gradients; the line weight is uniform; the

shapes are geometric and the faces are reduced to three marks; there is visible paper texture. Now you have a process description you can build with, and the client usually agrees it captures what they meant.

This is better on every axis. It is more reliable technically, because process terms are stable across seeds. It avoids trading commercially on a named person's reputation. And it produces something that belongs to the project rather than being a weak copy of someone whose actual work is better.

If they insist on the name, that is a conversation about the brief rather than about prompting, and the business block later gives you the language for it.

BEFORE YOU MOVE ON

Why does "risograph print, two colours, slight misregistration" reproduce a look more consistently than naming a contemporary illustrator?

A. Because process terms are shorter, so they occupy fewer tokens and leave more of the prompt window available for describing the subject.

B. Because artist names are increasingly filtered or down-weighted by model providers, so the prompt is often not reaching the model in the form it was typed.

C. Because every risograph print in the training data shares the same physical characteristics, while an artist's name covers decades of varied work and imitations.

D. Because process terms describe the image surface rather than its content, and surface attributes are applied at the end of denoising where they are less likely to be overridden.

The answer and why: p. 397

Lesson 26 · 7 min

Sixteen cheap frames beat four expensive ones

THE ONE THING TO KEEP

Judging a grid of many low-cost renders at once is faster and more accurate than judging full-quality images one at a time, because composition is visible at thumbnail size and commitment bias is not.

Borrowed from film, and still correct

A photographer shooting film did not examine each frame. They printed a contact sheet — every frame from the roll on one sheet, thumbnail-sized — and marked the ones worth enlarging. The reason it worked is that **composition is legible at thumbnail size and detail is not**, so looking at small images forces you to judge the thing that actually decides whether a frame is any good.

The same method, applied here, is the biggest single improvement most people can make to how they work.

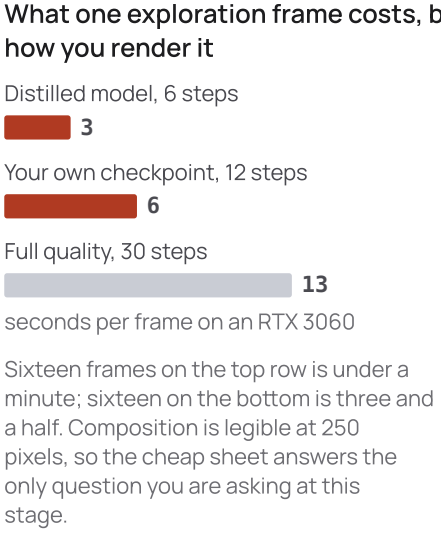
The method

1. Pick a fast, cheap configuration: a distilled model at 6 steps, or your normal model at 12–15 steps. Quality does not matter. Composition does.
2. Generate 16 images at the native ratio with random seeds, batched.
3. Assemble them as a 4 × 4 grid. Most interfaces have a grid output option; otherwise it is one line of ImageMagick, which is free.
4. Look at the grid **at thumbnail size**, not full screen.
5. Mark two or three. Discard the rest without sentiment.
6. Take the marked seeds forward at full quality.

```
montage *.png -tile 4x4 -geometry 256x256+4+4 sheet.jpg
```

On a mid-range card with a distilled model, sixteen frames is under a minute. On a hosted service at $0.01 a small image, it is sixteen cents. Either way it is cheap enough to do four times with different prompts, which is the point.

Figure 3.2



The three questions

Looking at the grid, ask only these:

1. **Does the eye land in the right place?** If you cannot tell what the picture is about at 250 pixels, no amount of finishing will fix it.
2. **Is the light doing something?** Flat frames stay flat.
3. **Is the frame structurally sound** — subject not cut awkwardly, not duplicated, room where the type goes?

Notice what is not on the list: hands, faces, small text, fine detail. All of those are repairable and none should influence the choice. The most expensive mistake in this work is choosing a frame with beautiful hands and mediocre composition, because you can fix the hands.

Why it beats iterating one at a time

Three reasons, and the third is the one people underrate.

Cost per decision. Sixteen cheap renders cost what two full-quality ones do, and give you eight times the information about composition.

Comparison rather than evaluation. Asked "is this good?", people say yes. Asked "which of these sixteen is best?", people answer accurately. Judgement is far more reliable when it is relative.

No commitment bias. After waiting ninety seconds for a full-quality render, you want it to be good, and you will start negotiating with yourself about its composition. A thumbnail you waited three seconds for gets thrown away honestly.

The cost of the cheap render

Being straight about the trade: a distilled model at 6 steps does not compose identically to the full model at 30. It has less variety, and it pulls toward the most probable interpretation of the prompt, so your sheet under-represents the unusual frames the full model would have produced.

Two things follow. First, if the sheet is dull, suspect the model before the prompt — run four at full quality to check. Second, use the *same* model at low steps when the difference matters; 12 steps on your real checkpoint is still four times cheaper than 30 and composes the way the final render will.

The more subtle failure is that a grid rewards frames that read instantly. That is exactly right for a thumbnail, a feed or a poster seen in passing, and it quietly penalises the quieter image that rewards a long look. If the deliverable is something people will sit with, mark one frame from the sheet that you suspect is better than it looks small, and render it properly before deciding.

Sheets of variations, not just of seeds

Once the habit is established, the sheet becomes the tool for every question, not only for seed selection.

- **Four prompts × four seeds** on one sheet tells you which phrasing is actually stronger, rather than which seed was lucky.
- **One prompt × four checkpoints × four seeds** tells you which model suits the brief, which is a question people otherwise answer from forum reputation.
- **One seed × a sweep of CFG values** shows you the burn point in a single glance.

The rule that makes all of these readable is to vary one thing along each axis and label it. A sheet where four things changed at once is a pretty picture and tells you nothing, which is the subject of a later lesson in this block.

Show the sheet to the client

An unexpected use, and a good one. Handing over a contact sheet with three frames circled turns an approval into a choice, and choices are easy for people who find open-ended feedback hard.

Two cautions. Only include frames you would be content to deliver, because somebody will pick the one you were least keen on. And say plainly that these are rough exploration renders, or you will get notes about hands.

BEFORE YOU MOVE ON

Why judge exploration frames as thumbnails on a grid rather than at full size one at a time?

A. Because thumbnails load faster, which allows many more candidates to be reviewed within the same working session.

B. Because small size hides repairable detail and forces judgement onto composition, and relative comparison is more reliable than absolute.

C. Because low-step renders are too unfinished to be judged at full size, so viewing them small avoids drawing conclusions from artefacts that the final render will not contain.

D. Because a grid layout makes it easier to spot duplicated subjects and other structural failures that appear across several frames at once.

The answer and why: p. 397

Lesson 27 · 7 min

Fix the seed or learn nothing

THE ONE THING TO KEEP

A seed fixes the starting noise, so changing it and the prompt together means you cannot attribute the difference to either, and seed-locked comparison is what turns guessing into testing.

The noise you started from

Generation begins with a field of random noise, and the seed is the number that produces it. Same seed, same everything else, same image. Change the seed and you have a different starting point, which produces a different picture even with an identical prompt.

That makes the seed the single most useful experimental control you have, and most people leave it on random and wonder why they cannot tell whether a prompt change helped.

The two modes

Random seed is for exploration. You are sampling the space of what this prompt can produce. Contact sheets run on random seeds by definition.

Fixed seed is for testing. You have a frame with the composition you want, and you are asking what one specific change does. Lock the seed, change one thing, compare.

Moving between them deliberately is the whole discipline. Explore on random. Test on fixed. Never test on random and then form a belief about the prompt, which is what nearly everyone does and is why the internet is full of prompt advice that does not replicate.

Figure 3.3

Two modes, and the discipline is moving between them on purpose

Random seed — for exploring

- Sampling what this prompt can produce
- Contact sheets run on it by definition
- Tells you nothing about a prompt change
- Forming a belief here is how folklore is made

Fixed seed — for testing

- One variable moves, everything else held
- Answers what that clause actually did
- A variation seed at 0.1 to 0.3 gives near neighbours
- Broken by resolution, sampler, interface or CPU noise

Seed 418 at 832 × 1216 and seed 418 at 1024 × 1024 are unrelated pictures, not the same one cropped, because the noise field has a different shape. Decide the generation size before you hunt.

Seed travel

Once you have a good seed, its neighbours are worth a look. On most interfaces this is "variation seed" with a "variation strength": the starting noise is blended between your seed and another, so at strength 0.1–0.3 you get the same composition with small differences — a hand that lands better, a slightly different expression, a shadow that falls more cleanly.

This is the cheapest improvement available. You already have the frame; you are asking for the same frame again with the dice rolled slightly. On a portrait where the face is nearly right, four variations at strength 0.15 will usually contain one that is materially better.

Sequential seeds are not neighbours, incidentally. Seed 1001 has nothing to do with seed 1000; the generator scatters them. Only the blend mechanism gives you actual nearness.

Write them down

The seed is in the PNG metadata, which is why the last lesson of the first block insisted on keeping PNGs. But a working log is faster to search than a folder of files:

```
2026-03-14  courtyard hero
  sheet A  prompt v3, 16 random, 12 steps     -> keep 418, 992
  418      full quality 30 steps              -> hands bad,
composition right
  418      +variation 0.15 x4                 -> 418/var2 best,
take forward
  992      full quality                       -> light flatter,
drop
```

Four lines. It is the difference between "I had a good one earlier" and opening it.

What a seed does not guarantee

A seed is only reproducible within the same configuration, and the list of things that break it is longer than people expect: the model, the sampler, the scheduler, the step count on ancestral samplers, the resolution, the interface version, and whether noise is generated on CPU or GPU.

That last one catches people constantly. Some interfaces default to CPU noise generation for cross-machine consistency and others use the GPU, and the same seed produces different noise between them. If a seed from somebody else's post produces a different image on your machine with every parameter matched, this is usually why, and it is a setting rather than a fault.

Resolution deserves its own warning. Seed 418 at 832 × 1216 and seed 418 at 1024 × 1024 are unrelated images, not the same image cropped, because the noise field has a different shape. So decide your generation resolution before

you start hunting for seeds. Finding a perfect frame and then discovering it was at the wrong size means starting the hunt again, and it is an entirely self-inflicted hour.

Seeds are not portable, and that is fine

A seed that somebody posts is worth almost nothing to you unless they also posted the model hash, the sampler, the scheduler, the step count, the resolution, the interface and its version. Miss any of those and the number produces a different picture.

This disappoints people who treat a good seed as a shareable asset. It is not one. What is shareable is everything around it — the prompt structure, the lighting description, the control image, the settings — and those transfer perfectly.

The right mental model is that a seed is a *bookmark in your own setup*, not a coordinate in a shared space. It is enormously valuable inside one project on one machine and meaningless outside it. Which is another argument for the environment file from the first block: the seed is only as durable as the configuration it points into.

The one habit worth building

At the end of a session, before closing anything, copy the two or three best seeds into the project log with a word about what each one is. It takes fifteen seconds. Without it you will have two hundred PNGs named `00042-3948573.png` and a vague memory that one of them was good, and finding it takes longer than generating sixteen more.

BEFORE YOU MOVE ON

Someone finds a good composition at seed 418, then switches from 832 × 1216 to 1024 × 1024 at the same seed and gets an unrelated image. Why?

A. The aspect ratio changed, so the model selected a different training bucket with a different set of compositional conventions attached to it.

B. Seeds are only valid within a single session, and changing any generation parameter causes the interface to reinitialise its underlying random state.

C. Larger images consume more random numbers, so the sequence drawn from the seed diverges partway through the construction of the noise field.

D. The seed builds a noise field shaped to the output size, so a different resolution is a different starting field.

The answer and why: p. 398

Lesson 28 · 7 min

Change one thing, or you have learned nothing

THE ONE THING TO KEEP

A comparison is only informative when one variable moves, and the habit of changing prompt, sampler and steps together is why so much accumulated knowledge about image generation turns out to be wrong.

Where bad advice comes from

Somebody changes their prompt, switches sampler, bumps steps from 20 to 35, and gets a better image. They post that DPM++ 3M SDE is better for portraits. Two thousand people repeat it. Nobody has tested it.

This is how almost all folklore in this field is produced, and it is why you should test the claims in this course too. The method is not complicated; it is just unglamorous.

The procedure

1. Fix the seed.
2. Fix everything else.
3. Change exactly one thing.
4. Generate both.
5. Look at them side by side at 100%.

Most interfaces have a built-in grid tool for this — A1111 and Forge call it X/Y/Z plot, ComfyUI does it with a primitive node in increment mode. It renders every combination on one sheet. A grid of 5 CFG values by 4 samplers is 20 images and answers the question properly in one run.

Things worth testing once, on your own work

Not endlessly. Once, and then write down the answer and stop revisiting it:

- **CFG**: render 3, 5, 7, 9, 12 at one seed. You are looking for where it starts to burn — over-saturated, hard-edged, crunchy. That is your ceiling. It differs per checkpoint and is usually lower than people run.
- **Steps**: 15, 25, 40, 60. Find where you stop seeing change. Usually around 25–30.
- **Samplers**: four of them at your settled CFG and steps. The differences will be smaller than you expect, which is itself the useful finding.
- **Quality tags**: with and against. Usually nothing.
- **Negative prompt**: your usual boilerplate against an empty one. Often surprisingly little.

That last one tends to change how people work. Long inherited negative prompts — the strings of "worst quality, jpeg artifacts, extra limbs, bad anatomy" — were tuned for particular SD 1.5 fine-tunes and frequently do

nothing on a modern checkpoint. Some models want an empty negative. You will not find out by reading; you will find out in four minutes with a fixed seed.

Why people do not do this

Because it feels slow, and because the exploratory mode — changing several things and seeing what happens — is genuinely enjoyable and occasionally productive.

Both modes are legitimate, and the point is to know which you are in. Playing is fine. Playing and then forming a firm belief about what caused the improvement is not. If a change seems to help, spend four minutes confirming it with a fixed seed before it enters your working practice, because an untested belief will cost you that four minutes every week for a year.

The limit of single-variable testing

Parameters interact. CFG and steps interact. A sampler's quality depends on the step count. A LoRA's right strength depends on the checkpoint. So a one-at-a-time test tells you about that variable *at your current settings*, and the answer can move when something else moves.

This is not a reason to abandon the method; it is a reason to re-test the two or three things that matter when you change checkpoint, and to prefer a small two-dimensional grid — CFG against steps, say — over two separate one-dimensional runs when you suspect interaction. Five by four is twenty images and perhaps three minutes on a distilled model.

The honest summary: you are not doing science, and you do not need to. You need enough rigour that your working defaults are true, and that is a handful of grids run once per checkpoint, written in a file next to your notes.

What the file should say

Keep it short enough that you actually maintain it. One block per checkpoint:

```
checkpoint: realvis-xl-v5  (hash a1b2c3d4)
  cfg:       5.5   (burns past 8)
  steps:     26    (no change past 30)
  sampler:   dpmpp_2m / karras
  negative:  empty — boilerplate made no difference
  buckets:   holds to 12:5; duplicates past that
  notes:     warm bias, grade cooler by ~200K at finish
  tested:    2026-03-14
```

That block took twenty minutes to produce and removes about forty decisions a week. The date matters, because when you update the interface or the checkpoint you need to know the numbers are stale.

Testing the advice you are given

One last use for this, including on this course. Everything written here about step counts, CFG ranges, denoise values and negative prompts is a starting point drawn from common behaviour across common models, and your checkpoint may differ.

So treat every number in this course as a hypothesis with a four-minute test attached. If your model burns at CFG 6 rather than 8, the number in your file is right and the one in the lesson is wrong for you. That is not a failure of the lesson; it is the method working.

BEFORE YOU MOVE ON

Why is a fixed seed necessary when testing whether a prompt change helped?

A. Because random seeds produce a wider spread of composition quality, which makes it harder to see small improvements against the natural variation.

B. Because interfaces cache the previous generation when the seed is unchanged, so the comparison runs faster and stays within one session.

C. Because a different seed produces a different image regardless of the prompt, so any difference cannot be attributed to the change.

D. Because some seeds are inherently better suited to certain prompts, and testing across seeds mixes prompt quality with an unrelated compatibility effect.

The answer and why: p. 398

Lesson 29 · 7 min

Recognising the twenty minutes that will never converge

THE ONE THING TO KEEP

Some requests fail for structural reasons rather than for want of a better prompt, and the professional skill is recognising them inside three attempts and switching method instead of continuing.

The sunk-cost hour

You have been at it for forty minutes. Each attempt is nearly right. One more phrasing will do it. It will not, and you knew by attempt four.

Certain requests do not respond to prompting at all, and the reason is always the same shape: the model is producing a plausible image, and plausibility does not include the constraint you care about. Recognising the category takes three attempts. Here are the categories.

The list

Exact counts above about four. "Seven birds" gives you five, or nine. The model has no counting mechanism; number words shift a distribution over how many things appear. Below four it usually lands; above, it drifts. Fix: generate the scene without the count and composite, or accept whatever number appears.

Legible specific text. Newer models can produce short phrases reasonably often, and even those fail on unusual words, small sizes and long strings. Fix: set the type yourself over a clean plate. This is not a workaround; it is how the industry works anyway, and the repair block covers it.

Precise spatial relations between three or more objects. "The red cup to the left of the blue book, behind the lamp" binds unreliably. Fix: sketch it and condition on the sketch, which is the whole of the next block.

A specific real object. Your client's actual bottle, with its actual label. The model has never seen it. Fix: photograph and composite.

A named living person's likeness. Blocked on hosted services, unreliable locally, and a separate legal question covered later. Fix: do not, unless you have permission and a real reason.

Consistent identity across many images from prompts alone. Fix: adapter or LoRA, which the consistency block covers.

Perfect symmetry or exact mechanical geometry. Logos, technical diagrams, repeating grids, architectural plans. Fix: vector tools. Inkscape is free and does in five minutes what forty prompts cannot.

Anatomy in unusual configurations. Hands interacting with objects, several people touching, a body from a rare angle. Fix: simpler pose, or repair, or pose conditioning.

The three-attempt rule

Adopt a rule and hold yourself to it: **three attempts, then change method.**

Attempt one, the straightforward prompt. Attempt two, rephrased with the failure addressed directly. Attempt three, one structural change — different model, much simpler framing. If all three fail in the same way, it is not a prompting problem. The identical failure repeating is the diagnostic: random failures suggest you are close, and a consistent one says the model cannot represent what you want.

What "change method" means

The alternatives are the rest of this course, and they are all stronger tools than a fourth prompt:

- Condition on an image — sketch, depth, pose, edges.
- Generate the parts separately and composite.
- Photograph the difficult element.
- Make it in a different kind of software entirely — Inkscape for vectors, Blender for geometry, a text tool for text.
- Change the brief so the impossible thing is not in frame.

That last one deserves more respect than it gets. If the composition requires a hand holding a fork at an unusual angle, and nothing works, a version where the hand is out of frame is often a better picture anyway. Constraints have improved compositions since long before any of this existed.

The honest caveat

This list has a shelf life. Text rendering was impossible, then unreliable, and is now workable on several models for short phrases. Counting and spatial binding have both improved with newer architectures. Anything written today about what models cannot do will be partly wrong within a year, and you should treat the list as a starting point rather than a law.

What does not change is the method: three attempts, watch for the *same* failure repeating, then switch. That rule stays correct regardless of which items are on the list this year, and it is the actual skill.

Telling the difference between close and stuck

The distinction that matters is between a *random* failure and a *systematic* one, and you can usually read it from three attempts.

Random failure — each attempt fails differently. The hands are wrong in one, the light is flat in another, the composition is off in the third. This means the model can produce what you want and you are sampling around it. Keep going, or run a contact sheet.

Systematic failure — each attempt fails the same way. Always the wrong number. Always the label illegible. Always the same anatomical impossibility at the same joint. This means the thing you want is outside what the model represents, and the twentieth attempt will look like the third.

Write down the failure after each attempt, in three words. Doing so converts a vague sense of frustration into a readable pattern, and the pattern answers the question.

Say it out loud on a shared job

If you are working with a client or a team, name it early: "This element is not going to come out of the generator — I'll shoot it and composite." Said on day one, that is expertise. Said on day four after a silent struggle, it reads as an excuse, even though it is the same sentence and the same fact.

BEFORE YOU MOVE ON

Three attempts at "seven birds on a wire" give five, nine and four birds. What does the consistency of the failure tell you?

A. That the prompt is ambiguous about arrangement, and specifying the spacing along the wire would allow the model to resolve the count correctly.

B. That the seed is unlucky, and sampling considerably more seeds will eventually produce a frame containing exactly seven of them.

C. That quantity is represented as a distribution rather than counted, so no rephrasing fixes it and the number has to come from compositing.

D. That the guidance scale is too low for the numeric part of the prompt to be enforced against the rest of the description around it.

The answer and why: p. 399

Lesson 30 · 7 min

Interrogating a picture for vocabulary

THE ONE THING TO KEEP

An interrogator recovers usable vocabulary from a reference image but never recovers the image, because it is describing appearance rather than inverting the generation process.

What the tools do

Point a CLIP interrogator or a tagger at a photograph and it returns text: a caption, or a list of tags. The free options are the CLIP Interrogator, the WD tagger family, and any general vision model you can run locally or through a free tier.

What comes back looks like this:

```
a woman in a green sari sorting chillies, sitting on the floor,
whitewashed wall, harsh sunlight, shadows, documentary
photography,
35mm, film grain, warm tones, india, natural light
```

Useful. And it will not regenerate the photograph.

Why it cannot give you the image back

An interrogator is a *description* model, not an inversion. It answers "what words fit this picture" by comparing the image against text embeddings and returning close matches. That is a different operation from recovering the specific noise, seed, model and prompt that produced a particular image — which, for a photograph, never existed anyway.

So the description is lossy in exactly the way descriptions always are. It captures category and salient attributes and loses arrangement, proportion, the precise light, the specific gesture. Feed the output back into a generator and you get *an* image of a woman sorting chillies in sunlight. Not that one.

People discover this and conclude the tool is broken. It is doing what it does; the expectation was wrong.

What it is genuinely good for

Recovering vocabulary you lack. You have a reference whose look you cannot name. The interrogator says `chiaroscuro`, `rim lighting`, `tenebrism` — and now you have three terms to use deliberately. This is the main value and it is a real one, especially early on.

Checking your assumptions. You think the reference is softly lit. The tagger says hard light. Look again; it is usually right, and your eye is being fooled by a bright background.

Building a caption style for LoRA training. When you caption a training set, consistency matters more than accuracy, and a tagger applied to every image gives you consistency for free. The consistency block returns to this.

Bulk sorting. Tag two hundred images and you can group them by style, which is how you build a reference library rather than a folder.

How to use the output properly

Never paste it in whole. It arrives full of redundancy and noise — `best quality`, `1girl`, duplicated concepts, and terms that came from the tagger's training vocabulary rather than from the image.

Read it, take the two or three words you did not have, and put them in the appropriate slot of your own structured prompt. The interrogator is a thesaurus, not a source.

The thing to be careful about

If the reference is somebody's copyrighted work, an interrogated prompt fed straight back is an attempt to reproduce it. Whether that is lawful depends on jurisdiction, on how close the result is, and on questions that are genuinely unresolved and being argued in several courts right now. This course does not pretend to know the answer.

What can be said is narrower and more useful: taking vocabulary from a reference — `hard side light`, `muted palette`, `35mm` — is ordinary practice, the same as any art director describing a look. Attempting to regenerate a specific existing picture is a different act with different exposure, and the difference is obvious to you when you are doing it.

A final practical note. The interrogator's output tells you what the *tagger* sees, which reflects its own training data and biases. It will describe people in its own vocabulary, which is sometimes crude and sometimes worse, particularly about ethnicity and body type. Read the output before you use it rather than pasting it forward unseen, because whatever it hands you becomes your prompt and then becomes your picture.

A better use: interrogate your own output

Turn the tool around. Run it on an image *you* generated and compare what it says against what you asked for.

If your prompt said `hard side light` and the interrogator says `soft studio lighting`, one of two things is true: the model ignored that part of the prompt, or your eye is wrong about what came out. Either is worth knowing, and both are invisible without the comparison.

This is a surprisingly good calibration exercise in the first month. Generate ten images, interrogate all ten, and read the captions against your prompts. The slots that consistently fail to appear in the description are the ones the model is not acting on, and they are where your prompt-writing time should go.

The free options, concretely

- **CLIP Interrogator** — runs on Hugging Face Spaces without installing anything, or locally on a modest GPU.
- **WD taggers** — tag-style output, fast, well suited to bulk captioning for training sets.
- **A small local vision-language model** — the most flexible option, because you can ask a specific question rather than accepting a generic caption. "Describe only the lighting in this image" is far more useful than a full caption when lighting is what you are after.

That last approach is the one worth learning. A general caption answers a question you did not ask; a targeted question answers the one you did.

BEFORE YOU MOVE ON

Why does feeding an interrogator's caption back into a generator not reproduce the source image?

A. Because interrogators are trained on generated images rather than photographs, so their captions transfer poorly to real-world source material.

B. Because the caption omits the seed and model parameters, which are what actually determine the specific output of a generation.

C. Because interrogators return tags rather than full sentences, and the loss of grammar removes the relationships between the elements described.

D. Because a caption describes appearance rather than inverting the generation, so arrangement, proportion and exact light are never captured.

The answer and why: p. 399

Lesson 31 · 7 min

What "good enough to take forward" actually means

THE ONE THING TO KEEP

A frame is worth repairing when its composition, light and structure are right, because those are the things repair cannot create, while hands, small objects and texture always can be fixed.

The decision people get wrong

At the end of exploration you have sixteen frames and have to choose. The instinct is to pick the one with the fewest visible faults. That is the wrong criterion, and it costs whole afternoons.

The right question is: **which faults can repair fix, and which can it not?** Choose on the second category and ignore the first entirely.

Cannot be fixed by repair

These decide the choice, because fixing them means generating again:

- **Composition.** Where things sit in the frame, the balance, how the eye moves. Inpainting works region by region; it cannot restructure a picture.
- **Light direction and quality.** You can grade colour and lift shadows. You cannot move a light source. A flatly lit frame stays flat.
- **Pose and gesture.** The whole attitude of a body. Repairing a badly posed figure means replacing it, which is a new generation.
- **Perspective and camera position.** Baked into every part of the frame at once.
- **Whether the mood is right.** Sounds soft, but it is the most expensive to be wrong about, because it is why the client will reject it.

Can be fixed, so ignore them now

- **Hands.** Always fixable. Mask and regenerate, or composite. Never reject a good composition for hands.
- **Faces at small size.** Fixable with a face-targeted repair pass.
- **Small objects, jewellery, buttons, cutlery.** Inpainting.
- **Text.** Never generated; always set afterwards.
- **Background clutter.** Removal or regeneration of that region.
- **Colour grade, contrast, warmth.** Editing, at the very end.
- **Resolution and sharpness.** Upscaling.
- **Minor anatomical oddities.** Repair.

The checklist

Run it on your two or three marked frames at 100%:

1. Does the composition work at thumbnail size? *(if no, discard)*

2. Is the light directional and consistent — do all shadows agree? *(if no, discard)*
3. Is the pose right, without repair? *(if no, discard)*
4. Is the crop-safe core intact for every required ratio? *(if no, can outpainting add room? if not, discard)*
5. Are there structural impossibilities — a limb emerging from the wrong place, a chair with three legs, a wall that does not meet the floor? *(if yes, discard)*
6. Everything else: list it as repair work and estimate the minutes.

If a frame passes one to five, it is a keeper regardless of how rough it looks. If it fails any of them, discard it without negotiation, no matter how good the rest is.

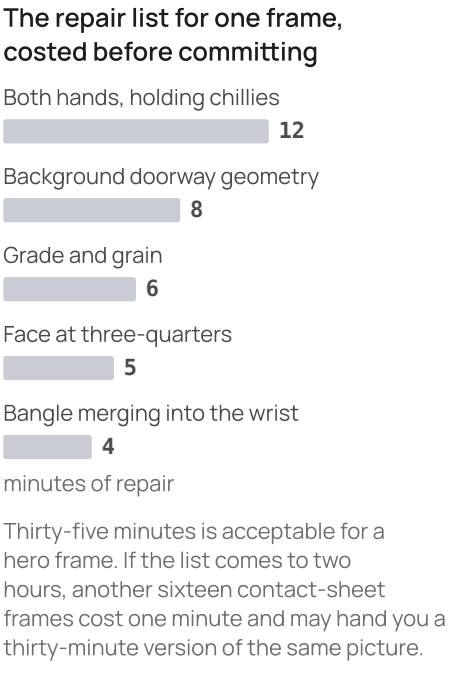
Counting the repair cost before you commit

Item six is not decoration. Write the list out and put a number against each:

```
seed 418
  hands, both, holding chillies        12 min
  face at 3/4, slight distortion        5 min
  bangle merges into wrist              4 min
  background doorway has odd geometry   8 min
  grade + grain                         6 min
                                       35 min
```

Thirty-five minutes is acceptable. If the list comes to two hours, go back to the contact sheet — another sixteen frames costs one minute and may hand you a thirty-minute version of the same picture. People almost never do this, and it is the single largest avoidable time loss in the whole process.

Figure 3.4



The honest tension

This checklist is conservative, and conservative choices produce competent, slightly safe work. A frame that fails item five with a strange impossible geometry is sometimes the most interesting image on the sheet, and the rule says discard it.

So hold the rule for commissioned work with a deadline, where predictability is what you are being paid for, and break it deliberately for your own work, where the strange frame is the point. What you should not do is break it by accident at 4pm on delivery day, discover at 6pm that the composition was never going to hold, and have no fallback. Knowing which mode you are in is the difference between a considered risk and an avoidable crisis.

Always keep a second frame

Take two forward, not one. The second is insurance and it costs almost nothing at this stage.

The reason is that some frames do not survive repair. You mask the hands, regenerate, and the new hands are fine but the wrist now reads oddly. You fix the wrist and the sleeve changes. Forty minutes later the image is worse than when you started and you have lost the version you liked, because you were working in place.

Two protections. Save the base render untouched before any repair, always, so you can restart. And keep a second candidate that you have not touched, so that if the first one unravels you have somewhere to go that is not the contact sheet.

Saying no to a frame the client likes

It happens: they pick the one that fails item two. The useful move is not to argue about taste but to name the consequence — "that one is lit flat, so it will look soft next to the others in the set, and I can't fix that after the fact." Then offer the one you would choose alongside it, rendered properly.

Presented with both at full quality, people usually change their minds, because the thing that was invisible on a contact sheet is obvious at full size. If they do not, you have said it once, in writing, and you build the flat one.

BEFORE YOU MOVE ON

Two candidate frames: one has excellent composition and badly mangled hands, the other has perfect hands and a flat, centred composition. Which goes forward?

A. The second, since correcting composition through cropping and outpainting is more predictable than repairing anatomy at full resolution.

B. Neither, because a frame requiring substantial repair in either category indicates the prompt needs revising before any candidate is taken forward.

C. The first, because composition cannot be created in repair while hands reliably can.

D. The first, but only if the hands can be replaced by compositing photographed hands rather than by regenerating them through inpainting.

The answer and why: p. 400

CASE STUDY · MODULE 3

Forty minutes on seven boats

An illustrative case, built from documented patterns.

A freelance designer in Kochi, producing a single hero image for a backwater tourism operator's spring campaign.

The brief was agreed and unusually clear: one hero frame, 1200 by 1500, a line of houseboats moored at dawn, mist on the water, no people, room for a headline across the top third. The operator had asked for one specific thing — seven boats, because the fleet is seven boats and the number appears in the copy.

Ramya started properly. Slots, not a sentence. Medium, subject, action, setting, light, camera, finish. She fixed the seed once she had a composition she liked and ran the first test.

Six boats.

She added *seven houseboats* at the front of the prompt, where CLIP weights earlier tokens more heavily. Nine boats. She tried *exactly seven wooden houseboats in a row*. Five. She tried weighting the number. She tried spelling it as a numeral. She moved to a FLUX-family model on the theory that a T5 encoder handling longer natural language might bind the count. Eight. She went back, raised the guidance, lowered it, changed sampler, and generated another batch.

Forty minutes had gone. Every attempt was a competent, attractive picture of houseboats at dawn with the wrong number of them. She had two hours before she wanted to send a first look.

This is the moment the lesson calls the sunk-cost hour, and the diagnostic was sitting in front of her if she had been writing it down. Each failure was the same failure. Not a random scatter of problems — flat light in one, bad composition in another, a duplicated hull in a third — but the identical defect every time, at every setting, on two different architectures. A random failure

means the model can produce what you want and you are sampling around it. A systematic one means the thing you want is outside what the model represents, and the twentieth attempt will look like the third.

Counting is a known systematic failure. There is no counting mechanism in a diffusion model. A number word shifts a distribution over how many things appear; below about four it usually lands and above it drifts.

Her options, once she named it, were three, and they had different costs.

Keep prompting. Cost: the remaining two hours, with no reason to expect a different outcome, and a first look that might not exist.

Change the brief. Frame the shot so only three boats are in view, receding into mist, with the rest implied. Cost: the number seven, which was the one thing the client had actually specified, and a conversation about why.

Change method. Generate the scene without a count at all, generate a single boat cleanly on its own, and composite the seven. Cost: perhaps ninety minutes of cutting out, scaling, matching perspective, painting reflections into the water, and matching grain — none of it glamorous, all of it predictable.

The third option also carried a second-order benefit she noticed only when she thought about it: a composited fleet is reusable. If the operator buys an eighth boat, that is a ten-minute change rather than a new hunt for a seed.

She also had one cheaper thing to do first, which she nearly skipped. She had never run a contact sheet on this brief. She had been judging one full-quality frame at a time for forty minutes, at ninety seconds a render, with commitment bias pulling her toward each one. Sixteen frames at twelve steps on her own checkpoint was ninety seconds in total.

YOUR ANSWERS

1. What distinguishes a random failure from a systematic one, and how would Ramya have spotted hers after three attempts rather than fifteen?

2. She changed to a FLUX-family model partway through. Was that a reasonable third attempt, and what did it tell her?

3. Why was the contact sheet worth running even after she had already decided to composite?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 166.

CASE STUDY · MODULE 3

How it played out

She ran the sheet, at a wider ratio than she needed so the crop had slack. Two of the sixteen had a stronger composition than anything she had produced in the previous forty minutes, and one of them had the mist doing something the prompt had never described. She took that seed forward at full quality, then composited: the base plate had three boats, she generated a single clean boat on plain water, cut it out four times at descending scale, and painted the reflections in GIMP with a soft brush at low opacity. Total, including repair and grade, was a hundred minutes. The count is exact, because she placed it. The operator's only note was that the mist was perfect. Ramya now keeps a three-word failure log beside her — she writes what went wrong after each attempt — because it was the writing down, not the knowing, that she had been missing: the pattern was visible after attempt three and she did not see it until attempt fifteen.

The questions, discussed

1. What distinguishes a random failure from a systematic one, and how would Ramya have spotted hers after three attempts rather than fifteen?

A random failure is different each time — bad hands here, flat light there — which means the model can produce what you want and you are sampling around it. A systematic failure is the identical defect at every setting, which means the thing you want is outside what the model represents. Writing the failure down in three words after each attempt turns a vague sense of frustration into a readable pattern.

2. She changed to a FLUX-family model partway through. Was that a reasonable third attempt, and what did it tell her?

Yes — the three-attempt rule allows one structural change, and a different text encoder is a genuine structural change rather than another rephrasing. The useful part is what it told her: the same failure on a different architecture is strong evidence that the problem is the mechanism, not the prompt.

3. Why was the contact sheet worth running even after she had already decided to composite?

Because she had been choosing a base plate by evaluating single full-quality frames, which is both slower and less accurate than comparing many cheap ones. Composition is legible at thumbnail size and commitment bias is not, so ninety seconds of sixteen frames gave her a better base than forty minutes of one-at-a-time judgement had.

MODULE 3 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. 394. If two go wrong, re-read the module before the exam.

1 A ninety-word prompt on an SDXL checkpoint puts all the right things in none of the right relationships. What is the mechanism?

A. The prompt is encoded in 77-token chunks with no attention across the boundary.
B. The latent has too few channels to hold that many concepts at once.
C. The sampler discards tokens beyond the number of steps it has budget for.
D. Longer prompts lower the effective guidance scale, so every clause is applied more weakly.

2 Somebody tells you that adding "8k, masterpiece, highly detailed" improves a modern checkpoint. How do you settle it, and what is the likely finding?

A. Compare against a second checkpoint, since the tags are model-specific.
B. Raise the guidance scale alongside the tags so that they carry more weight.
C. Generate twenty images with the tags and judge the average quality.
D. Fix the seed, generate with and without, and expect almost no difference.

3 Why does leaving light unspecified produce the recognisably "generated" look?

A. Checkpoint merges bake in a glossy house look that no amount of prompting can remove.
B. Unspecified light returns the model's average, which has no dominant source.
C. Diffusion models compute light physically and default to ambient occlusion.
D. The VAE lifts shadows during decoding unless a lighting term is present.

4 Why is "risograph, two colours, slight misregistration" more stable across seeds than naming a living illustrator?

A. Artist names are stripped by most hosted services before the prompt is encoded.
B. Process terms are shorter, so they sit comfortably inside the first fifteen tokens where weighting is strongest.
C. Every risograph print in the training data shares the same physical signature.
D. Style names only work on models that were fine-tuned on that artist's work.

5 Which fault should not influence which frame you take forward from a contact sheet?

A. A duplicated subject caused by an extreme ratio.
B. A subject cut awkwardly by the frame edge.
C. Flat light with no direction anywhere in the frame.
D. Hands with the wrong number of fingers.

6 Somebody posts a seed and a prompt, you match both, and your render is a different picture. What is the usual explanation?

A. Something in the configuration differs — sampler, resolution, or CPU against GPU noise.

B. Their image was upscaled, which rewrites the seed recorded in the metadata.

C. Seeds expire once the provider supersedes that model version.

D. Sequential seeds drift, so seed 1000 and seed 1001 give related but slightly different noise.

4

MODULE 4

Steering with pictures instead of adjectives

Take control of composition by conditioning on an image — a photograph, a biro sketch, a flat colour block, a depth render or a pose skeleton — choose the conditioning type that preserves what the brief requires and discards what it does not, and state the strength and end-step you used and the failure each setting was avoiding.

32 Stop describing it and show it the picture 172
33 The model is guessing your composition 176
34 Three passes down the denoise ladder 179
35 A biro sketch is the most reliable composition tool you own 184
36 Paint flat shapes, then let the model make them real 187
37 Get the perspective right by building it 191
38 Canny, lineart, depth, softedge, pose, normal 194
39 Different words for different parts of the frame 198
40 Transferring a look from an image without training anything 202
41 Two controls, without the plastic look 206
Case study: The hour of Blender nobody wanted to spend 211
Module 4 test 216

Lesson 32 · 8 min

Stop describing it and show it the picture

THE ONE THING TO KEEP

Denoise strength decides how much of your reference survives; the useful range is roughly 0.4 to 0.6, and everything above 0.75 throws the reference away.

Adjectives are a bad way to specify a layout

You want a shot of your uncle's tailoring shop: the counter on the left, the fabric rolls stacked to the ceiling on the right, the doorway blown out with daylight. You can write four hundred words about it and the model will still put the counter wherever it likes, because you are describing a picture to something that matches captions rather than reads floor plans.

You have a phone. Take the photograph. Then hand the photograph to the model.

This is the hinge of the whole course. Almost every technique from here on is a way of giving the model something other than words.

Denoise strength is the only dial that matters

Image-to-image works like this: instead of starting from pure noise, the model starts from *your* image with a controlled amount of noise mixed in, then denoises from there. How much noise it adds is the strength — sometimes labelled denoise, denoising strength, or image weight.

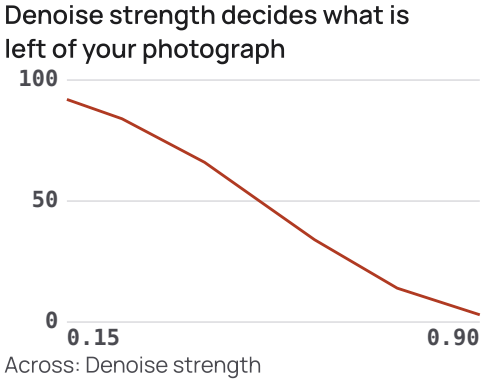
That one number decides how much of your picture survives.

- **0.15-0.25** — your image back, slightly repainted. Useful for a texture or style tweak, useless for changing content.
- **0.4-0.6** — the working range. Same composition, same masses of light and dark, genuinely different rendering. This is where most useful img2img lives.

- **0.75 and up** — the model has effectively been handed static. You get something loosely inspired by your image and you have wasted the reference.

People try 0.8, see nothing of their photo, conclude img2img does not work, and go back to writing adjectives. The range is narrow and you find it by bisecting: run 0.3, 0.5 and 0.7 on the same seed, look at all three, then split the interval that looked closest.

Figure 4.1



Across: Denoise strength
Up: Reference surviving, %
—— Roughly how much of your input is still there

The working range is 0.4 to 0.6: same composition, same masses of light and dark, genuinely different rendering. At 0.8 the model has been handed static and you have wasted the reference.

Three different things called "reference"

Tools confuse these and it causes real frustration.

- **Structure reference** — keep the layout, change everything else. This is img2img at mid strength, and it is what you want for the tailoring shop.
- **Style reference** — take the palette, the brushwork, the grade, and apply it to a different subject. Midjourney exposes this as `--sref`; hosted tools usually call it a style image.

- **Character or subject reference** — carry a specific face or object into a new scene. Midjourney's `--cref`, and the identity features covered in lesson six. This is the hardest of the three and the one most likely to disappoint.

If a tool gives you one box marked "reference image", find out which of the three it is doing before you blame your input.

What ruins a reference

Resolution and aspect ratio. A 400-pixel WhatsApp forward carries compression blocks, and at mid strength the model will faithfully reproduce blockiness as texture. Shoot at full resolution. Match the reference's aspect ratio to your output, or the model crops or stretches in ways you did not ask for.

Busy backgrounds. The model has no idea which parts of your reference you care about. If the fabric rolls matter and the pile of boxes does not, crop the boxes out before you upload. A cleaner reference is a stronger instruction.

Text in the reference. It will come back as almost-words. Cover it or crop it, and add real type later.

Where this leaves you, and where it does not

Img2img gives you a composition you chose rather than one the model guessed. It does not give you precise control: at 0.5 strength the doorway is still in the doorway's place, but the number of fabric rolls will change and the counter's proportions will drift. If you need the geometry to hold exactly — a pose matched frame to frame, a room's perspective preserved — that is conditioning, and it is the next lesson.

Free path, all of it: **Krita with the AI Diffusion plugin** gives you a canvas, a strength slider and local generation, free. **ComfyUI** and **InvokeAI** the same. On a phone, most hosted tools accept an uploaded image in their app, and the Gemini app in particular is built around editing a picture you give it.

The consent line, briefly

A style reference taken from one living artist's work, used to produce commercial output in their manner, is not a technical setting. It is a decision about somebody's livelihood, and courts in several countries are currently arguing about it. Referencing a movement, an era or a printing process is a different act from referencing a named person still selling work. Lesson ten returns to this.

Today: photograph something in your own room, run it through any img2img tool at 0.3, 0.5 and 0.7 with the same prompt and seed, and put the three results side by side. You now know that dial for that tool, which is knowledge no article can give you.

BEFORE YOU MOVE ON

Someone feeds a phone photo of their shop into an image-to-image tool at 0.85 strength and gets a picture with nothing of the shop in it. They drop to 0.15 and get their own photo back with faint grain. What is going on?

A. Strength sets how much noise is added before denoising begins, so it controls how much of the original survives; both values sit at the extremes, and the range that keeps the layout while changing the rendering is around the middle, found by bisecting.

B. The reference is too low-resolution for the model to read its layout, so it falls back on the prompt.

C. Image-to-image transfers style but never composition; only pose or depth conditioning can hold a layout.

D. The prompt is competing with the reference and needs to be shortened until the image can come through.

The answer and why: p. 402

Lesson 33 · 9 min

The model is guessing your composition

THE ONE THING TO KEEP

Conditioning makes composition an input rather than an outcome, and letting the control end around 60% of the steps is what stops the result looking traced.

Every generation is a guess about where things go

The prompt says a woman reaching upward for a book on a high shelf. The model produces a woman, a shelf and some books, and arranges them however the training data suggests. Run it again and the arrangement changes. You are sampling from a distribution of plausible compositions, not specifying one.

Conditioning changes that. Instead of hoping, you hand the model the composition directly.

The mechanism, in one paragraph

Take a reference image and run it through a preprocessor that throws away almost everything and keeps one kind of structure: a stick figure of the joints, a greyscale near-and-far map, a map of edges. A second network reads that structure and, at every denoising step, nudges the image toward matching it. The prompt still decides what things look like. The map decides where they are.

This is the ControlNet family, released in 2023 by Lvmin Zhang, and it is the single biggest reason working practitioners use ComfyUI or Forge rather than a hosted box.

Which map for which job

- **OpenPose** — a skeleton of body, hand and face keypoints. Use it when the pose is the thing and nothing else about the reference should carry over. You can pose a stick figure by hand and generate from that, with no source photograph at all.
- **Depth** — a greyscale image where near is bright and far is dark. Use it to keep a three-dimensional layout while changing everything's appearance: same room, different decade. The best choice for architecture, interiors and putting a product into a scene.
- **Canny** — a strict edge map. Keeps every line it finds. Use for turning a clean line drawing into a rendered image.
- **Lineart** and **Softedge** — looser edge detection. Softedge in particular gives the model room to reinterpret while keeping the shape.
- **Scribble** — deliberately crude. Draw with a mouse in thirty seconds, get a composition.
- **Segmentation** — regions labelled by category: sky here, building there, road there.

The two settings that decide whether it looks alive

Control weight. At 1.0 the conditioning is enforced hard. Drop to 0.6-0.8 and the model has room to make the image work as an image.

Start and end percent. This is the one people miss. You can apply the control for only part of the denoising process. Composition is decided early; detail and texture are decided late. So run the control from 0% to about 60% of the steps and then let go. The layout is locked in by then, and the model spends the final steps resolving the picture naturally instead of tracing.

An image conditioned at weight 1.0 for 100% of the steps looks stiff, flat and traced, because it is. Ending the control early is the difference between a render and a colouring-in exercise.

The failure modes

Stacked controls fighting each other. Pose plus depth plus canny, all at full weight, leaves the model no freedom and no way to satisfy all three. The result is muddy and rigid. Use one control. Add a second only when you can name what it is contributing.

Canny eating artefacts. Canny finds edges, and JPEG compression blocks, paper grain and scanner noise all look like edges. Run canny on a compressed photo and you get thin phantom lines in the sky. Raise the canny thresholds, or clean the source first, or use softedge instead.

Depth flattening. Depth maps from a photo with a plain background often read the whole background as one distance, so your generated scene has a wall where you wanted a street.

Where hosted tools stand

Some expose a sketch-to-image or pose-reference feature that is conditioning under a friendlier name. Most do not expose it at all, and none of them give you the weight and scheduling controls. If your work needs a pose held across a series, this is usually the moment you install something local.

The free path is the whole path

There is no paid version of this that is better. The ControlNet models are free downloads on Hugging Face. **ComfyUI** and **Forge** expose every setting. **Krita with AI Diffusion** exposes them in a way that will not frighten you, with a canvas you can draw the scribble on directly. Free GPU hours on Colab or Kaggle run all of it. See running models yourself.

Today: draw a stick figure or a five-line scribble, run it through scribble or openpose conditioning at weight 0.8 ending at 60%, and watch a composition you chose come back rendered. That is the moment most people stop treating generation as a slot machine.

BEFORE YOU MOVE ON

A designer conditions a render on a scanned pencil sketch using Canny edges at weight 1.0 for the full run. The result looks stiff and flat, and there are thin stray lines across the sky. What went wrong?

A. Canny is the wrong preprocessor for drawings and depth conditioning should be used instead.

B. Canny keeps every edge it detects, including paper grain and scan artefacts, and holding the control at full weight for all the steps prevents the model from resolving the drawing into a rendered image; raising the thresholds and ending the control around 60% fixes both.

C. The sketch was scanned at too high a resolution, so far too many edges were detected.

D. The run needs more denoising steps so the model has time to smooth the traced lines away.

The answer and why: p. 402

Lesson 34 · 8 min

Three passes down the denoise ladder

THE ONE THING TO KEEP

Running several img2img passes at decreasing denoise strength converges on a finished image more reliably than one pass at a middling value, because each pass only has to make a small correction.

One big jump, or three small ones

The usual way people use img2img is a single pass: load the reference, pick a denoise value, generate, look, adjust the number, repeat. It works, and it is worse than the alternative for a reason that is easy to see once stated.

At a single middling value — 0.55, say — the model has to do two jobs at once: reorganise the image substantially, and finish it convincingly. It is being asked to make a large change and a refined one in the same pass, and one of them suffers. Either the composition survives and the surface stays rough, or the surface is beautiful and the composition has drifted.

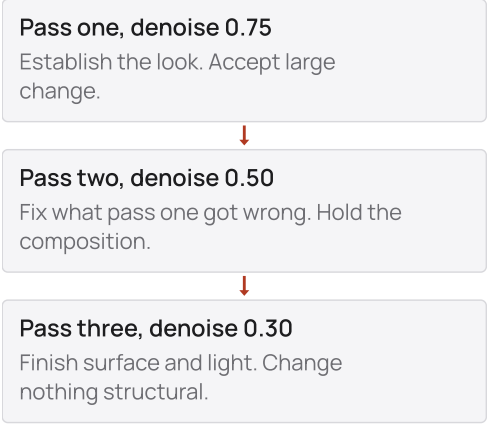
The ladder splits the jobs.

```
pass 1   denoise 0.75   establish the look, accept large change
pass 2   denoise 0.50   fix what pass 1 got wrong, hold
composition
pass 3   denoise 0.30   finish surface and light, change nothing
structural
```

Each pass takes its predecessor as input. Same prompt, same seed or not, only the strength moves.

Figure 4.2

Three rungs, each starting from something better



One pass at 0.55 asks for a large change and a refined one at the same time, and one of them suffers. Correct the compounded colour and contrast drift once, at the end, rather than on every rung.

What the numbers actually do

Denoise strength decides how far back toward pure noise the image is pushed before denoising begins. At 1.0 nothing of the input survives; at 0.0 nothing changes. Between those, the number is roughly the proportion of the sampling schedule that runs.

Useful landmarks, which hold across most models:

- **0.15–0.25** — texture and grain only. The picture is identical.
- **0.30–0.40** — surfaces, light quality, small detail. Composition untouched.
- **0.45–0.60** — the working range. Real change, recognisable source. Objects can move slightly, expressions change, materials change completely.
- **0.65–0.75** — the source is now a suggestion. Composition survives loosely, content may not.
- **0.80+** — the input is contributing little beyond a colour distribution.

The ladder works because 0.30 applied to a nearly-right image is a completely different operation from 0.30 applied to a rough one. Each rung starts from something better than the last did.

Where this is the right tool

Bringing a rough render up. A sketch, a 3D blockout, a crude collage. Start at 0.7 to make it photographic, then descend.

Changing a material. Concrete to timber, plastic to ceramic. Two passes at 0.5 and 0.3 with the material named in the prompt.

Matching a look across a set. Take frame nine, run it at 0.3 with the anchor's prompt, and its light and grade move toward the rest of the set without its content moving.

Rescuing a frame that is nearly there. One pass at 0.35 often repairs the general softness that makes an image feel unfinished.

The way it goes wrong

Colour drift. Each pass nudges the palette, usually warmer and more saturated, because each pass re-imposes the model's own bias. Three passes compound it. Check the grade after the ladder and correct it once at the end rather than fighting it on every rung.

Contrast crush. The same compounding applies to contrast; blacks get blacker each time. If your third pass looks heavy, this is why.

Loss of the thing you liked. The reason you chose that frame was some specific quality, and three passes of averaging tends to sand it off. Keep the original open in another window and compare, not just the previous rung.

Detail invention. At 0.5 and above the model will add things that were not there — a pattern on a wall, a reflection, an extra object. On a product shot this is a correctness problem, not a style one.

The limitation to understand

Img2img cannot move something across the frame. It has no notion of objects; it is denoising a whole latent toward a distribution consistent with the prompt. A chair on the left stays on the left, or becomes something else on the left.

So the ladder is the wrong tool for "move the subject right and give me more sky". That is outpainting, or a control image, or a new generation. When somebody spends forty minutes raising denoise to try to reposition a subject, this is what they have missed: by the time the strength is high enough to move it, the strength is also high enough that nothing else survives either.

The right division is that img2img changes *what things are made of and how they are lit*. Controls, which the rest of this block covers, change *where things are*. Reaching for the wrong one is the single commonest waste of time in this part of the work.

BEFORE YOU MOVE ON

A designer runs img2img at increasing denoise values trying to shift the subject from the centre to the left of the frame. By 0.8 the subject has moved but everything else has changed too. Why does this approach fail?

A. Img2img denoises the whole latent toward the prompt rather than manipulating objects, so position only changes when the source stops surviving.

B. The denoise value applies uniformly across the frame, so shifting a subject requires a masked pass at a higher strength on the target region than on the surrounding area.

C. The seed was not held constant between passes, so the composition was free to drift independently of the denoise strength being tested.

D. The model's attention layers bind the subject to the centre of the latent, and only a positional control signal such as a pose skeleton can override that binding.

The answer and why: p. 403

Lesson 35 · 8 min

A biro sketch is the most reliable composition tool you own

THE ONE THING TO KEEP

Five minutes of rough drawing photographed on a phone gives the model a composition it will follow, and it requires no drawing skill because the conditioning reads structure rather than quality of line.

The objection first

"I cannot draw." Almost nobody reading this can, and it does not matter. What the model needs from a sketch is where things are, how big they are, and roughly what shape. A shaky rectangle in the lower left is a table. Two circles and a line are a person. The conditioning model extracts structure, and structure survives bad drawing entirely.

This is worth taking seriously because it is the highest-leverage technique in this block, and the barrier to it is psychological rather than technical.

The method

1. **Draw it.** Paper, biro, five minutes. Or in Krita or GIMP with a mouse, which is free and just as good. Block shapes, not detail. Mark the horizon and where the light comes from.
2. **Photograph it** with a phone, flat and evenly lit, or scan it.
3. **Clean it up** — raise contrast so the lines are black on white. Thirty seconds with Levels in GIMP.
4. **Feed it as a control**, using scribble or lineart conditioning, at strength 0.6–0.8 and ending around 0.6–0.7 of the steps.
5. **Prompt normally**, describing the scene in the slot structure from the previous block.

The scribble control type exists precisely for this and is forgiving of rough input. Lineart is cleaner and stricter. Canny edge detection is the wrong choice for a sketch, because it will faithfully preserve every wobble in your line.

Why it beats prompting for composition

A prompt describes a composition in words the model interprets probabilistically. "A figure on the left, a doorway behind, the table in the foreground" is three spatial relationships, and spatial relationships are exactly what language conditioning binds worst. The previous block listed this as a structural failure: three or more objects in stated positions do not reliably land.

A sketch is not a description. It is the answer. The model does not have to infer the arrangement, because the arrangement is the input.

The practical consequence is that a five-minute sketch replaces about forty minutes of prompt attempts, and it lands the first time.

Where to let go

The single most common mistake is over-drawing. A detailed sketch at high control strength produces a result that looks exactly like a coloured-in drawing, because it is one. The model has been given no room to contribute anything.

So:

- Draw the **big shapes only**. Five to ten elements, no more.
- Mark **the horizon and the light direction** — these two lines carry an enormous amount.
- Leave faces, hands and detail completely blank. Draw an oval for a head.
- Use **strength 0.6–0.75, not 1.0**, and **end the control around 0.6–0.7** so the last third of the sampling is free to make it look real.

That end-step setting is the one people miss. Running a control to the final step means the structure is enforced right through the phase where fine texture is formed, and the result acquires a flat, traced quality. Releasing it

early gives you the composition you drew and a surface the model built.

The iteration this unlocks

Sketching changes the shape of the work, not just its accuracy. Instead of generating and judging, you draw three compositions in ten minutes, generate each, and compare. You are making compositional decisions on paper, where they take seconds and cost nothing, rather than in a generator where each attempt is a render and a wait.

This is how art direction has always worked — thumbnails first, then execution — and it transfers here without modification.

What a sketch cannot give you

It fixes arrangement. It does not fix perspective, and this catches people. If your drawn box does not converge correctly, the model will often follow your wrong perspective and produce a subtly impossible room that nobody can quite explain. The conditioning is faithful, including to your mistakes.

For anything where perspective must be correct — architecture, interiors, a product on a surface — the next tool in this block is the answer: build the blockout in three dimensions and let the software get the perspective right. A sketch is for arrangement. A depth render is for geometry. Knowing which problem you have saves you from fixing a perspective error by redrawing, which mostly produces a different perspective error.

BEFORE YOU MOVE ON

Why should a sketch control be set to end around 60–70% of the steps rather than running to completion?

A. Because control models lose accuracy in the final steps, so continuing past that point introduces structural drift rather than preventing it.

B. Because the final steps form fine texture, and enforcing the drawn structure through them makes the result look traced.

C. Because running a control to the end consumes additional memory in the later steps, where activation sizes are at their peak for the sampling process.

D. Because ending the control early allows the prompt to take over and correct any compositional errors present in the original sketch.

The answer and why: p. 403

Lesson 36 · 8 min

Paint flat shapes, then let the model make them real

THE ONE THING TO KEEP

A crude painting of flat colour shapes run through img2img at low denoise places objects and palette exactly, because the colour distribution of the input survives the parts of the schedule that decide layout.

The technique nobody teaches

Open GIMP or Krita. Make a canvas at your generation size. With the bucket and a fat brush, paint blocks: a band of grey-blue across the top for sky, a brown mass lower left for a table, a warm shape in the middle for a person, a pale rectangle for a window. Ten shapes, two minutes, no skill whatsoever.

Run that through img2img at denoise 0.55–0.7 with a proper prompt. You get a photographic image whose layout and palette match your blocks.

This is the fastest way to control **where things are and what colour they are** at the same time, and it is almost unknown outside people who have worked it out themselves.

Why it works

Diffusion resolves an image coarse-to-fine. The early steps of the schedule decide broad layout and colour distribution; the later steps fill in detail. When you start from an existing image, the early part of the schedule is skipped in proportion to the denoise strength — so at 0.6, roughly the first 40% of the schedule never runs, and the broad layout and colour decisions have already been made by your painting.

The model then does the remaining 60%: deciding what those shapes are made of, how they catch light, and what their surfaces look like.

This also explains the strength range. Below about 0.45 the model cannot make your flat shapes into convincing objects; they stay flat. Above about 0.75 your layout stops mattering. The window is roughly 0.5 to 0.7 and the best value depends on how rough your blocks are — cruder input wants a higher number.

What it is best at

Exact brand colour placement, following on from the brief block. Paint the blocks in the brand palette and the output arrives in that palette, needing only a small correction rather than a full regrade.

Layout with several elements. Four objects in specified positions, which prompts handle badly and blocks handle exactly.

Value structure — where the image is light and where it is dark. Paint in greyscale first if colour is confusing you. A strong light-and-dark arrangement is most of what makes a composition work, and it is much easier to judge in three tones than in colour.

Skies and gradients. A painted gradient survives and gives you the sky you wanted rather than the model's default.

Doing it well

- **Use the right hues, not the right objects.** The model reads a brown mass as something brown in that position. Do not try to draw a table.
- **Keep the value range honest.** If the scene is dark, paint it dark. Painting mid-grey everywhere gives you a flat result.
- **Blur it slightly.** A Gaussian blur of a few pixels over your blocks removes hard edges the model would otherwise try to preserve as literal boundaries.
- **Add a light gradient** across the whole canvas in the direction of your light source. This one trick makes the output dramatically more convincing, because you are handing the model a lighting structure as well as a layout.
- **Combine with a control** if you also need structure: colour blocks for palette, depth or scribble for geometry. The stacking lesson at the end of this block covers how.

The free toolchain

All of it works in software that costs nothing. **Krita** has the best brushes and is the most pleasant to block in. **GIMP** is fine and is probably already installed. **Photopea** runs in a browser, which matters if you are on a borrowed or locked-down machine. You need the bucket fill, a large soft brush, and Gaussian blur. Nothing else.

Where it disappoints

Colour blocking controls layout and palette. It does not control **identity or fine form**. A brown mass becomes a table, or a cardboard box, or a low wall — whatever the prompt suggests and the shape permits. If you need specifically a table, say so in the prompt and accept that its design is the model's choice.

It also struggles with **thin structures**. A painted line for a lamp stand is usually too thin to survive; the denoising absorbs it into the background. Anything narrower than roughly 1% of the frame width needs either a control image or a repair pass afterwards.

And there is a characteristic failure worth recognising: at the lower end of the strength range, the output keeps a faint flatness, as though a poster had been photographed. That is the blocks not being fully transformed, and the fix is one more pass at 0.35 rather than a higher strength on the first, which would cost you the layout you painted.

BEFORE YOU MOVE ON

Why does a flat colour-block painting run at denoise 0.6 control the layout of the output?

A. Because flat regions of uniform colour contain little high-frequency detail, so the sampler has less existing structure to work against in those areas.

B. Because the early part of the schedule is skipped, so broad layout and colour are already fixed by the painting when denoising begins.

C. Because the model treats the input image as a conditioning signal alongside the text prompt, weighting the two according to the denoise value that has been set.

D. Because colour information survives the encode to latent space more faithfully than spatial detail does, so palette persists while layout is regenerated.

The answer and why: p. 404

Lesson 37 · 9 min

Get the perspective right by building it

THE ONE THING TO KEEP

Rendering a depth map from a few primitives in Blender gives the model correct perspective and object placement that no sketch or prompt can supply, and it is the standard professional route for interiors, architecture and product staging.

The problem a sketch cannot solve

Draw a room. Unless you know how to construct perspective, your walls will not converge correctly, and the model will faithfully reproduce your error. The result is a space that feels wrong in a way viewers notice without being able to name — and clients notice it precisely when the image is on a large screen in a meeting.

The fix is to stop drawing perspective and start building it. Software does perspective perfectly, for free, and you need about an hour of Blender to use it.

What you actually build

Not a model. A **blockout**: grey boxes standing in for things.

1. Open Blender. Delete the default cube or keep it as your first object.
2. Add planes for floor and walls. Add cubes for furniture, boxes, a counter. A cylinder for a bottle, a rough figure for a person — a capsule is enough.
3. Position the camera. This is the part that matters: set the focal length to the value you would have written in the prompt, and set the height to a realistic eye or table level.
4. Render a **depth pass** — in the Compositor, enable the Z pass, normalise it, and output it. Or use the Mist pass, which is easier and often good enough.

You now have a greyscale image where near is light and far is dark. That is a depth map, and it is exactly what depth conditioning consumes.

Why this is worth the hour

Perspective is correct by construction. The vanishing points are right because the software computed them.

The camera is a real camera. A 35mm lens at 1.2 metres produces the actual geometry of that setup, not an approximation of the word.

You can move it. Four angles of the same scene, all geometrically consistent, in two minutes. Try that with a sketch.

Scale is honest. Objects are the right size relative to each other, which is something generated images get subtly wrong all the time — the mug that is slightly too large, the chair that could not seat anyone.

It is reusable. The blockout is a file. Next month's variation on the same scene starts from it.

Running it

Feed the depth map as a depth control at strength 0.5–0.8, ending around 0.7 of the steps. Then prompt the scene normally: materials, light, mood, finish.

Depth control preserves **spatial arrangement and form** and discards everything else. Your grey boxes become timber, marble, upholstery — whatever the prompt says — while staying exactly where and how big you placed them.

A practical detail: keep the depth range tight. If your scene spans two metres but the camera can see a distant wall fifty metres away, normalisation squashes everything of interest into a narrow band of grey and the control becomes weak. Either move the far clipping plane in, or use Mist with a manually set start and depth.

Where else this applies

- **Product staging.** A cylinder where the bottle goes, correct height, correct camera. Generate the environment around it, then composite the real product into the space you reserved.
- **Architecture and interiors.** The whole point. A rough massing model renders into a photographic visual.
- **Consistent sets.** Twelve images of the same room from different angles, geometrically consistent, which is a problem the consistency block otherwise cannot solve.
- **Repeatable client revisions.** "Move the sofa" is a thirty-second change in the blockout and a re-render, rather than a new hunt for a seed.

The honest costs

Blender is a serious application and its interface is not friendly on day one. What you need here is genuinely small — add a primitive, move it, set the camera, render a pass — and that is perhaps an hour of a tutorial. But an hour is an hour, and if you need one image this week it is not the right tool.

Depth is ambiguous about material. A depth map says nothing about what anything is made of, so if the prompt is vague the model will decide, and it will decide differently on every seed. Depth controls space, the prompt controls substance, and both have to be specified.

Depth discards edges. Two objects at the same distance blend into one grey region and may merge in the output. When that matters, stack a second control — the last lesson of this block — or separate them in depth by moving one slightly.

There is also a free alternative worth knowing about if Blender is too much: run a depth estimator over an existing photograph. A monocular depth model will produce a usable depth map from any picture, and it is one click in most interfaces. It gives you the perspective of a scene you photographed rather than one you designed, which covers a surprising number of briefs.

BEFORE YOU MOVE ON

What does depth conditioning from a Blender blockout provide that a hand-drawn sketch cannot?

A. Geometrically correct perspective and true relative scale, because the camera projection is computed rather than estimated by hand.

B. A stronger conditioning signal, since depth maps are continuous greyscale while sketches are binary line art and therefore carry less information per pixel.

C. Control over materials and surface finish, because the renderer assigns a surface property to every object in the blockout before the depth pass is written out.

D. Separation between objects that sit at the same distance from the camera, which line art merges into a single undifferentiated region.

The answer and why: p. 404

Lesson 38 · 8 min

Canny, lineart, depth, softedge, pose, normal

THE ONE THING TO KEEP

Each control type preserves one property of the input and discards the rest, so the choice is made by naming what must survive from the source and what must be free to change.

One question decides it

Not "which control is best". **What must survive from my input, and what must be free to change?**

Answer that and the control picks itself.

The types, and what each keeps

Canny detects hard edges and produces a thin black-and-white line image. It preserves *every* edge, including texture edges, noise and the grain of a wall. Extremely faithful and correspondingly rigid. Use it when the source is clean and you want the output to match it closely — a product outline, a logo shape, an architectural photograph you are restyling. Its threshold settings matter enormously: too low and you capture noise, too high and you lose the object.

Lineart is a learned edge extractor that produces drawing-like lines. It ignores texture and keeps object boundaries. Better than Canny for anything organic, and the right choice for turning a clean drawing into an image.

Scribble is the loosest. Trained on rough, wobbly, human lines. Forgiving of bad drawing, which is exactly what you want from a five-minute sketch.

Softedge (HED and its relatives) sits between lineart and depth — soft, thick boundaries that convey form without dictating exact contours. It is the most forgiving control for photographic sources and my default when I want "roughly this arrangement" rather than "exactly this shape".

Depth keeps distance and volume, discards edges, texture and colour. The right tool for spatial arrangement and perspective, as the previous lesson covered.

Normal encodes surface orientation — which way each bit of a surface is facing. It preserves three-dimensional form and how light will fall on it, more finely than depth. Useful for detailed objects, faces and anything where the sculptural quality matters.

OpenPose extracts a skeleton: joints and limbs, nothing else. It keeps a body's position and gesture and discards body shape, clothing, age and identity entirely. Unmatched for posing a figure, and useless for anything that is not a body.

Segmentation labels regions by category — sky here, building there, road below. Keeps layout at the level of what-kind-of-thing-goes-where, discards all form. Good for landscape and urban scenes.

Figure 4.3

Each control keeps one property and discards the rest

Canny — every hard edge
Keeps texture, grain and JPEG blocks too. Rigid.

Lineart — object boundaries
Ignores texture. Right for a clean drawing.

Scribble — rough human lines
Forgiving of a five-minute biro sketch.

Softedge — thick soft boundaries
Roughly this arrangement, not exactly this shape.

Depth — distance and volume
Discards edges, texture and colour entirely.

Normal — surface orientation
Sculptural form, and how light will fall on it.

OpenPose — joints and limbs
Discards body shape, clothing, age, identity.

Segmentation — regions by category
Sky here, building there. All form discarded.

The question is never which is best. Name what must survive from the input and what must be free to change, and the control picks itself.

Matching a job to a type

- Turn my sketch into a photograph → **scribble**
- Restyle this photograph but keep the exact objects → **canny** or **lineart**
- Keep this room's geometry, change everything about its look → **depth**

- Put a different person in this exact pose → **openpose**
- Keep this figure's pose and the room's layout → **openpose** plus **depth**
- Keep the sculptural form of this object → **normal**
- Keep the general arrangement, let details change → **softedge**
- Recolour a product but hold its silhouette exactly → **canny**, high strength

The two settings that matter more than the type

Strength — how hard the control pushes, usually 0 to 2 with 1.0 as neutral. Below 0.5 it is a suggestion; above 1.3 it dominates and the output starts to look like a coloured-in control image. The working range is 0.6 to 1.0 for most jobs.

Start and end steps — when the control is active. Ending it at 0.6–0.8 of the schedule is right for most work, for the reason the sketch lesson gave: the last portion of sampling should be free to build texture. Starting late (around 0.2) is an unusual but useful trick when you want the model to establish its own composition and then be nudged toward your structure.

People reach for a different control type when the real problem is that strength is at 1.0 and should be at 0.7. Change the settings before the type.

The failure mode of every control

All of them share one: **too much control produces something that looks constructed.** Flat, posed, slightly dead, with the subject's outline just a bit too exactly matching the input.

The cause is straightforward. The generative process has degrees of freedom, and each constraint removes some. Constrain the edges, the depth, the pose and the colour and there is very little left for the model to decide, so the output stops having the incidental richness that makes an image look real — the small asymmetries, the details nobody specified.

The practical rule that follows: **constrain only what the brief actually requires, at the lowest strength that holds it.** If the brief needs the pose, use pose and nothing else. Adding depth "for safety" costs you the

liveliness and buys you nothing the brief asked for.

BEFORE YOU MOVE ON

You need to put a different person into the exact pose of a reference photograph, with different clothing, build and setting. Which control type fits?

A. Canny, because the edge map captures the outline of the pose precisely and gives the strongest possible structural constraint.

B. Depth, because it preserves the three-dimensional arrangement of the limbs while leaving surface appearance free for the prompt to decide.

C. Softedge, because its thick soft boundaries convey the position of the figure without dictating the exact contours of the body.

D. OpenPose, because it extracts joint positions only and discards build, clothing and identity.

The answer and why: p. 405

Lesson 39 · 8 min

Different words for different parts of the frame

THE ONE THING TO KEEP

Regional prompting assigns separate text to separate areas of the image, which solves attribute binding failures that no amount of rephrasing a single prompt can fix.

The problem it solves

Two people, one in a blue coat and one in a red scarf. The model swaps them. You emphasise, reorder, add weights. It still swaps them, and the previous block explained why: attribute binding degrades, especially across chunk

boundaries and especially between similar subjects.

Regional prompting removes the ambiguity by removing the need to bind. Instead of one prompt describing two people, you give the left region one prompt and the right region another. There is nothing to confuse, because the descriptions never meet.

How it works

Implementations differ — ComfyUI has conditioning-area and mask nodes, A1111 and Forge have regional prompter extensions — but the mechanism is the same everywhere. The image is divided into regions, either by simple splits or by painted masks. Each region gets its own conditioning. During sampling, each region's prediction is computed with its own text embedding and composited according to the masks.

A simple setup:

```
base prompt (whole image, low weight)
  a documentary photograph, two people in a market, hard
sunlight

region left (0.0-0.5 width)
  an older man in a blue wool coat, grey beard

region right (0.5-1.0 width)
  a young woman in a red silk scarf, laughing
```

The coat stays blue. The scarf stays red. No emphasis syntax, no seed lottery.

What it is good for

Two or more subjects with distinct attributes. The canonical case.

Sky and ground treated separately. A dramatic sky over a specific landscape, each described properly, which one prompt handles poorly because the words compete.

A specific object in a specific place. "Red kettle on the left of the counter" as a region rather than a hope.

Keeping a character's description off the background. If your prompt says "woman in a green sari" and the background acquires a green tint, that is the description bleeding into the whole frame. Regions confine it.

Mixing styles deliberately. A photographic subject against an illustrated background, which is otherwise very hard.

The settings that decide whether it works

Base prompt weight. There is usually a global prompt applied across the whole image, controlling overall style and light. Keep it descriptive of the *scene* and let regions describe the *contents*. If the base is too strong it overrides regions; too weak and the image loses coherence. Around 0.2–0.4 of the total weight is a reasonable start.

Mask feathering. Hard region boundaries produce a visible seam — a straight vertical line where the style changes. Feather the masks by 30–60 pixels and the transition disappears.

Overlap. Regions that touch but do not overlap can leave a strip that belongs to neither. A small overlap is safer than a gap.

What it does not fix

It does not create objects. A region saying "a red kettle" on an empty counter produces a red kettle only if the composition already supports one there. Regional conditioning biases what appears in an area; it does not place objects. If the base composition has a window there, you will get a reddish window.

It does not handle interaction. Two people shaking hands cannot be split into two regions, because the hands are in both. Anything where subjects touch or overlap needs a different approach — generate them separately and composite, or accept a repair pass.

It struggles with irregular shapes. Rectangular splits are reliable. A complex painted mask around a figure that has not been generated yet is guessing at a shape that does not exist, and the result is often worse than not masking at all.

When to use something else

Regional prompting is fiddly to set up, and there is usually a simpler route. Before reaching for it, consider two alternatives.

Inpainting. Generate the scene, then mask the coat and regenerate it blue. Slower per image, far more reliable, and it works on any interface. For one image, this is usually the better answer.

Generate separately and composite. Two clean subjects on plain backgrounds, cut out and assembled over a generated environment. More work, complete control, and the standard approach for anything a client will scrutinise.

Regional prompting earns its complexity on **sets** — when you need twenty images with the same two-subject structure and cannot afford twenty inpainting sessions. For a single hero image, set it up only if you already know the tool; otherwise inpaint.

BEFORE YOU MOVE ON

Why does regional prompting solve attribute swapping between two subjects when prompt emphasis does not?

A. Because each region is sampled independently and composited afterwards, so the two descriptions never interact at any point in the process.

B. Because regional masks are applied in pixel space after decoding, which prevents the text encoder's attribute confusion from reaching the final image.

C. Because the attributes are described in separate conditioning applied to separate areas, so there is no binding to get wrong.

D. Because emphasis syntax multiplies token weights uniformly across the prompt, which strengthens both descriptions equally and therefore cannot resolve a conflict between them.

The answer and why: p. 405

Lesson 40 · 8 min

Transferring a look from an image without training anything

THE ONE THING TO KEEP

An image-prompt adapter injects a reference image's appearance through the model's attention layers, giving style transfer in seconds rather than the hours a LoRA costs, at the price of much less control over what exactly is transferred.

A third way to use a reference

You have three mechanisms for giving a model a picture, and they do different things.

- **img2img** starts from the image, so content and layout survive.
- **A control** extracts one structural property and enforces it.
- **An image-prompt adapter** — IP-Adapter is the common family — injects the image's *appearance* into the generation as though it were part of the prompt.

The third is the one people find hardest to place, and it is the one that answers "make it look like this, but of something else".

What it actually does

The adapter runs the reference through an image encoder and feeds the resulting features into the model's cross-attention layers, alongside the text. The model is then attending to a picture and a sentence at once.

The practical consequence is that it transfers a *blend* of everything the encoder noticed: palette, lighting, texture, mood, and — unless you take steps — composition and subject too. That blending is both the appeal and the weakness.

The variants worth knowing

Most implementations offer several modes, and choosing correctly matters more than tuning strength:

- **Standard** — transfers everything. Expect the subject to come along with the style.
- **Style only** — transfers palette, texture and light, leaves composition alone. This is the one you want most of the time.
- **Composition only** — the reverse; arrangement without appearance.
- **Face** variants — specialised for identity, covered in the consistency block.

Weight sits around 0.5–0.8 for a noticeable effect without domination. Above 1.0 the reference starts reproducing itself, which is both an aesthetic problem and, if the reference is somebody else's work, a different kind of problem.

Adapter or LoRA

The question comes up constantly and the answer is a decision rule, not a preference.

Use an adapter when: you have one reference, you need it now, you want an approximate look, and it is a one-off. Setup is seconds and there is nothing to train.

Use a LoRA when: you need the same look across many images over time, you need it precise, you have 15–30 example images, and you can spend an hour training. The result is stronger, more consistent and reusable.

The rough shape: an adapter gives you perhaps 70% of a look in 30 seconds; a LoRA gives you 95% in an hour. For a single image, take the 70%. For a campaign, train.

Using it well

One reference, not five. Stacking adapters averages them into something characterless.

Match the medium. A photographic reference for a photographic output. Mixing an illustration reference into a photographic prompt produces a muddled result that is neither.

Crop the reference to what you want. If you want the palette, crop to a region that is mostly the palette. The encoder attends to the whole image, so a reference containing a person will contribute a person.

Combine with a control for structure. The adapter handles look, a depth or pose control handles arrangement. This pairing is the workhorse combination for consistent sets.

The limitations, stated plainly

You cannot say which attribute to take. "Take the palette but not the composition" is only available as far as the mode switch allows. Unlike a slot-structured prompt, you have no fine control over what transfers. This is the

main frustration and it is inherent to the mechanism: the encoder produces one feature vector, not a labelled list of attributes.

It weakens fine prompt control. The image features compete with the text for attention, so detailed prompt instructions land less reliably when an adapter is running at high weight. If your prompt suddenly stops being followed, lower the adapter.

It can reproduce the reference too closely. At high weight with a distinctive source, the output can be recognisably derived from it. That is a practical risk when the reference is not yours, and the rights block returns to it. The working rule: if you can look at the output and identify the reference, the weight is too high for commercial work.

Quality depends on the encoder. These adapters are trained against a specific image encoder and base model. An adapter built for SD 1.5 will not work on SDXL, and mismatched combinations produce weak or bizarre results rather than clean errors.

BEFORE YOU MOVE ON

A designer needs one image in the visual style of a supplied reference, today. What makes an image-prompt adapter the right choice over training a LoRA?

A. An adapter transfers style through attention at generation time rather than through weight updates, so it applies more faithfully to a single unseen subject.

B. An adapter requires no dataset and no training time, which suits a one-off where approximate is enough.

C. An adapter can be adjusted after generation by changing its weight, whereas a LoRA's contribution to the output is fixed once the training run has finished.

D. An adapter avoids the licensing question a trained LoRA raises, since no derived weights are produced from the reference material at any stage.

The answer and why: p. 406

Lesson 41 · 8 min

Two controls, without the plastic look

THE ONE THING TO KEEP

Stacking conditioning signals removes the model's freedom in proportion to the total constraint, so the sum of the strengths matters more than any individual value and should stay near 1.2 to 1.5.

When one control is not enough

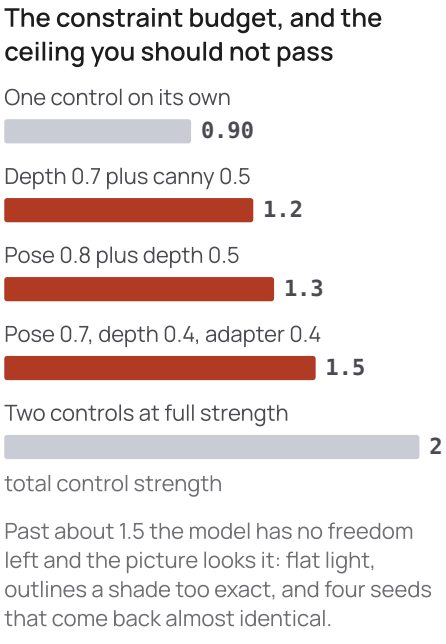
A figure in a specific pose, in a room with specific geometry. Pose control gets the body; depth gets the room. You need both, and both is where results start going wrong.

Two controls at strength 1.0 each is not "well controlled". It is a picture with almost no freedom left, and it looks it.

The budget

Treat total constraint as a budget of roughly **1.2 to 1.5** across all active controls. Some worked allocations:

Figure 4.4




```
one control                  0.8 - 1.0
pose 0.8  +  depth 0.5       = 1.3
depth 0.7 +  canny 0.5       = 1.2
pose 0.7  +  depth 0.4  +  ip-adapter 0.4   = 1.5, at the
ceiling
```

Also stagger the end steps. Running everything to the same point is unnecessarily rigid; a useful pattern is to let the structural control end earliest and the looser one run slightly longer:

```
depth   strength 0.6   end at 0.5    (geometry is settled early)
pose    strength 0.8   end at 0.7    (the body needs holding
longer)
```

Combinations that earn their complexity

Pose + depth — a person in a designed space. The most common pair in commercial work.

Depth + colour-block img2img — geometry from a blockout, palette from a painting. Run the blocks as the starting image at denoise 0.6 with the depth control active.

Canny + IP-Adapter style — keep a product's exact silhouette, change its entire material and setting. The best route to "the same bottle, ten different scenes".

Softedge + regional prompting — loose arrangement with attribute control per area.

Depth + normal — for objects where sculptural form matters and depth alone reads flat. Unusually, these two agree with each other, so they can run higher than most pairs.

When controls disagree

This is the real source of trouble. If your pose skeleton puts an arm where your depth map has a solid wall, the model has to satisfy two contradictory demands. What you get is an arm passing through the wall, a melted region where the two meet, or one control silently losing.

The failure is diagnosable: a specific area that looks wrong in every seed while everything else is fine. When you see that, overlay the two control images in an editor and look at that area. They will be arguing.

The fix is to make the inputs agree before generating, not to tune strengths afterwards. Move the wall in the blockout. Adjust the pose. Five minutes on the inputs beats an hour on the sliders, and tuning strengths on contradictory inputs only decides which constraint wins rather than removing the conflict.

The symptom of over-control

Learn to recognise it, because it is easy to produce and hard to see if you have been staring at the image:

- Flat lighting despite a lighting prompt, because the model had no room to build form.
- Outlines slightly too crisp and too exactly matching the control.

- A posed, dead quality, as though everything were waiting.
- Backgrounds that are simple and uninteresting.
- Every seed looking nearly the same. This is the clearest signal of all: if four seeds are almost identical, you have over-constrained, because seed variation is exactly the freedom you removed.

Run that last check deliberately. Four seeds at your current settings. If they differ meaningfully, the constraint level is healthy. If they do not, drop the total by 0.3 and try again.

Knowing when to stop stacking

The deeper point is that stacking controls is often a sign that the job has moved past generation. If you need this pose, this geometry, this palette, this silhouette and this style, you have specified the image almost completely — and at that point building it from parts is usually faster and always more controllable.

Generate the elements separately, each with one control and plenty of freedom, and assemble them in a layered editor. That is compositing, it is the subject of the next block, and it is how most professional work involving these tools is actually produced. The control stack is for when you want one coherent render with a few things held fixed, not for when you want to specify everything.

BEFORE YOU MOVE ON

Four different seeds with pose and depth controls both at strength 1.0 produce nearly identical images. What does that indicate?

A. The controls are contradicting each other, which has collapsed the range of outputs that can satisfy both constraints at once.

B. The seeds are too close together numerically, so the starting noise fields are similar enough to produce comparable compositions.

C. Total constraint is too high, so almost no freedom is left for the seed to influence.

D. The control images are being applied to the full sampling schedule, so the seed's contribution is overwritten during the final steps of denoising.

The answer and why: p. 406

CASE STUDY · MODULE 4

The hour of Blender nobody wanted to spend

An illustrative case, built from documented patterns.

A three-person interiors practice in Jaipur, pitching a fit-out for a district co-operative bank branch.

The pitch needed four visuals of a branch interior that did not exist yet: the customer hall from the entrance, the same hall from the counter, a corner with the queuing system, and a manager's cabin. The practice had the floor plan, the furniture schedule and two weeks.

Sameer, who does their visuals, started with prompts. The results were beautiful and useless. Each one was a plausible bank interior; none of them was *this* bank interior, and no two agreed about where the counter was or how high the ceiling ran. When he described the layout in words — the counter along the left, a waiting bank of six chairs facing it, a glazed cabin at the back right — the model bound about half of it and invented the rest.

He moved to sketching, which is the right instinct and the cheaper tool. Five minutes of biro, photographed on a phone, contrast raised in GIMP, fed as a scribble control at strength 0.7 ending at 0.65. The arrangement came back correct. The counter was on the left. The chairs were where he had drawn them.

And the room was subtly impossible. His drawn walls did not converge on a consistent vanishing point, and the conditioning had been faithful to his mistake. The senior partner, who has looked at interiors for twenty years, said it felt wrong within four seconds of seeing it and could not say why for another thirty. That is the specific failure of a sketch: it fixes arrangement and it does not fix perspective, and clients notice the difference precisely when the image is large on a screen in a meeting.

So the decision, at the start of week two.

Redraw the sketches more carefully. Sameer can construct perspective by hand; he learned it at college. Cost: two hours per view, four views, and a real chance of producing a different perspective error rather than none,

because hand-constructed perspective is exactly as accurate as the person doing it at eleven at night.

Keep the sketch route and fix the geometry in repair. Cost: perspective is baked into every part of a frame at once. It is the category the keeper checklist puts under *cannot be fixed by repair*, and an hour spent discovering that is an hour gone.

Learn enough Blender to build a blockout. Not a model — grey boxes. Planes for floor and walls, cubes for the counter and the chairs, a cylinder for the queue post, a capsule where a person stands. Set a camera with a real focal length at a real eye height, render a depth pass, feed it as a depth control at 0.6 ending at 0.7, and prompt the materials and the light. Cost: an hour of a tutorial and an afternoon of fumbling, on a pitch with a deadline, by a person who had avoided Blender for six years.

The practice had one more consideration that decided it. The bank's committee would ask for changes. They always do, and the changes are always *move the counter* and *can we see more of the waiting area*. On the sketch route, each of those is a new drawing and a new hunt for a seed. On a blockout, moving the counter is thirty seconds and a re-render, and the four views stay geometrically consistent with each other because they are four cameras in one scene.

YOUR ANSWERS

1. The sketch produced the right arrangement and the wrong room. What exactly does a sketch control, and what does it not?

2. Why did the depth map nearly fail, and what was the fix?

3. What tipped the decision toward the blockout, given that it was the slowest option to start?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 214.

CASE STUDY · MODULE 4

How it played out

The Blender afternoon cost six hours rather than one, most of it lost to the depth pass being normalised across a far wall fifty metres away that the camera could see through a window, which squashed everything of interest into a narrow band of grey and made the control almost useless. Moving the far clipping plane in fixed it in two minutes once he understood what he was looking at. After that the four views took about forty minutes each, including repair, and the perspective was correct by construction because the software had computed it. The committee asked for two changes: move the counter two metres toward the entrance, and show more of the waiting area. Both were done in the blockout in under ten minutes and re-rendered the same afternoon, and all four views stayed consistent. The practice now keeps blockouts as project files, and the bank branch scene became the starting point for a credit-society fit-out three months later.

The questions, discussed

1. The sketch produced the right arrangement and the wrong room. What exactly does a sketch control, and what does it not?

A scribble or lineart control preserves structure — where things are, how big, roughly what shape — and it is faithful to the input including its errors. It does not construct perspective. If the drawn walls do not converge correctly, the model reproduces that, and the result is a space viewers feel is wrong without being able to name why.

2. Why did the depth map nearly fail, and what was the fix?

Depth is normalised across the whole visible range. With a wall fifty metres away in shot, the two metres of the room that actually mattered were compressed into a narrow band of grey, so the control carried almost no information. Moving the far clipping plane in — or using Mist with a manually set start and depth — restores the range where the scene is.

3. What tipped the decision toward the blockout, given that it was the slowest option to start?

The revisions. A committee asking to move the counter is a thirty-second change and a re-render on a blockout, and a new drawing plus a new seed hunt on the sketch route. The blockout is also a reusable file, and four views rendered from one scene are geometrically consistent with each other in a way four separate sketches never are.

MODULE 4 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. 401. If two go wrong, re-read the module before the exam.

1 You run img2img at 0.85 and almost nothing of your photograph survives. What should you conclude?

A. The reference was too low in resolution to carry structure at that strength.

B. Img2img needs a control image alongside it before anything is preserved.

C. The strength is far above the useful range; bisect between 0.3 and 0.7.

D. The reference's aspect ratio did not match the output size.

2 A conditioned image comes back flat and traced. Which setting is most likely wrong?

A. The control model is built for a different base and should be swapped.

B. The control runs to the final step instead of ending around 60 per cent.

C. The preprocessor ran at a lower resolution than the generation.

D. The negative prompt is empty, so nothing is steering away from stiffness.

3 Why does a two-minute painting of flat colour blocks control layout and palette at the same time?

A. Starting from an image skips the early schedule, where layout and colour are set.

B. The VAE encodes flat colour more accurately than it encodes photographic texture.

C. A low denoise disables the text encoder, so only the image is read.

D. The blocks are read as a segmentation map by the conditioning network, region by region.

4 You need a different person in exactly this photograph's pose, with everything else free to change. Which control?

A. Softedge, which keeps the arrangement without exact contours.

B. Canny at high strength, to hold the outline exactly.

C. Depth, because it keeps the figure's position and volume in the room.

D. OpenPose, which keeps joints and discards shape and clothing.

5 Four seeds at your current settings come back almost identical. What does that tell you?

A. The guidance scale is too high and has burned the variation out.

B. The sampler is non-ancestral and the image has fully converged by this step count.

C. You have over-constrained; seed variation is the freedom you removed.

D. The checkpoint is a merge and has collapsed toward its house look.

6 One area of a conditioned image looks wrong on every seed while everything else is fine. What is the diagnosis and the fix?

A. That region falls outside the model's trained bucket; crop and regenerate.

B. Two controls disagree there; make the inputs agree before generating.

C. The mask feathering is too tight; widen it by thirty to sixty pixels.

D. The preprocessor failed there; re-run it at a higher resolution.

5

MODULE 5

Repair, extend, assemble

Take a flawed base render to a deliverable by masked repair — fix hands, faces and small objects at full resolution with stated denoise values, extend the frame without a visible seam, remove an object cleanly, and composite a photographed item into a generated scene with matching perspective, contact shadow, optics and grain.

42 The finished image is never one generation 220
43 Fix it in the order that stops you doing it twice 224
44 The three repairs everybody needs, with settings 229
45 Type is set, not generated 233
46 Extending a frame when the crop changes 238
47 Four ways to remove an object, and when each is right 242
48 The real product in a generated world 245
49 Making two sources agree 250
50 The fault the client sees on a big screen 254
51 Getting to poster size without a second horizon 257
Case study: A 4:5 frame, a 16:9 banner, and two hours 262
Module 5 test 268

Lesson 42 · 9 min

The finished image is never one generation

THE ONE THING TO KEEP

Generate a base plate, then repair it region by region; the "inpaint at full resolution" setting is what gives a small detail enough pixels to be drawn correctly.

The gap between a good image and a usable one

The image is right. The expression is right, the light is right, the client will like it. And there are six fingers on the left hand, the shop sign says CAFFEE, and there is a stray object floating near the shoulder.

Beginners regenerate. They roll the dice again, lose the good expression, get worse hands, and spend an afternoon and a month's credits chasing a single perfect roll. Working illustrators do not do this. They accept the frame as a base plate and repair it.

Inpainting is that repair, and it is where most of the real work in this craft happens.

How inpainting works, and the one setting that matters

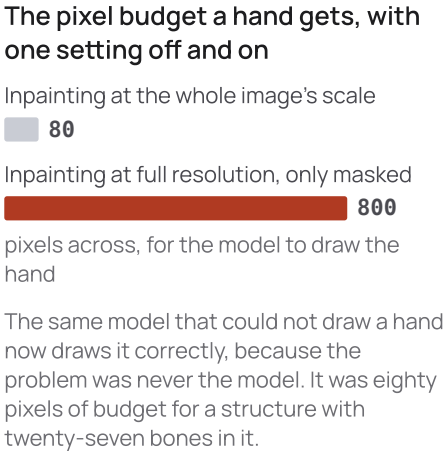
You paint a mask over a region. The model regenerates only inside the mask, conditioned on everything outside it, so the new pixels match the surrounding light and colour.

Now the setting that separates people who can fix hands from people who cannot. Look for a control called **"inpaint at full resolution"**, **"only masked"**, or **"crop to mask"**.

With it off, the model regenerates your masked region at the whole image's scale. If your picture is 1024 pixels wide and the hand occupies 80 of them, the model gets 80 pixels of budget to draw a hand. It cannot. It produces the same mush every time, and you conclude the model is bad at hands.

With it on, the tool crops to the masked area plus some padding, regenerates *that crop* at the model's full working resolution — so the hand gets 800 pixels of attention instead of 80 — and scales the result back in. The same model that could not draw a hand now draws it correctly, because the problem was never the model. It was the pixel budget.

Figure 5.1



That single toggle fixes more images than any prompt technique in existence.

The other three controls

Mask blur / feather. A hard mask edge leaves a visible seam. A few pixels of blur lets the new region fade into the old. Too much blur and the model starts blending the thing you were trying to remove back in.

Padding. How much surrounding context the crop includes. Too little and the model cannot see the arm the hand belongs to, so it draws a hand pointing the wrong way. Too much and you are back to a small pixel budget.

Denoise strength, again. For a repair, 0.6-0.8 — you want it to actually replace what is there. For a subtle correction, 0.3.

Outpainting, and the thing clients always ask for

Outpainting extends the canvas: the model invents new image beyond the existing edges, matched to what is there.

The everyday use is not artistic. You made a beautiful 16:9 frame and the client needs 9:16 for Stories and 1:1 for the feed. Cropping destroys the composition. Outpainting builds the missing top and bottom, and then you crop from a larger canvas.

Photoshop calls this Generative Expand. Free equivalents that do the same job: **Krita with AI Diffusion**, **InvokeAI's canvas**, **ComfyUI** with an outpaint node, and most hosted tools have an expand or uncrop button.

The failure modes, with their causes

Colour drift. Inpaint the same area eight times and the region slowly diverges in colour and contrast from the rest, because each pass is a fresh generation matched only approximately to its surroundings. Fix it outside the model: flatten, then a curves and colour-balance pass on that region in a raster editor. Do not try to inpaint your way back to matching.

Halo edges. Almost always too little padding or a mask drawn tight to the object's silhouette. Mask generously — include some of the background around what you are replacing.

The model ignores your mask. Some tools mask the *latent* rather than the pixels, which means the edges bleed. If a tool consistently modifies outside your mask, it is that, and switching tools is faster than fighting it.

The workflow this implies

1. Generate cheaply until you have a base plate with the right composition and mood. Stop there. Do not chase perfection.
2. Inpaint each defect separately, at full resolution, one at a time.
3. Flatten and do a colour and contrast pass in an editor.
4. Only then upscale.

That order matters and lesson seven explains why the last step is last.

Today: take any generated image you have with a flaw in it, open a free inpainting tool, mask the flaw, and confirm you have found the "only masked" setting before you press generate. If the tool does not have one, that tells you something about the tool.

BEFORE YOU MOVE ON

Someone repeatedly inpaints a mangled hand in a 1024-pixel-wide image and it stays mangled, even though the same model draws perfect hands when asked for a close-up of a hand. What is the most likely cause?

A. The model is genuinely weak at hands and only a hand-specific LoRA will fix it.

B. The mask is feathered too heavily, so the original hand is being blended back into the new one.

C. The prompt does not describe the hand's position, so the model has nothing to work from.

D. Without "only masked" or "inpaint at full resolution" enabled, the region is regenerated at the whole image's scale, so the hand gets a few dozen pixels of budget and there is no room for the detail.

The answer and why: p. 408

Lesson 43 · 8 min

Fix it in the order that stops you doing it twice

THE ONE THING TO KEEP

Repair has a correct order — structure, then faces and hands, then edges, then small objects, then grade and grain last — because each stage changes the pixels the next stage would otherwise have worked on.

Why order matters at all

Most people repair in the order they notice things. They see a bad hand, fix it. Then they notice the background doorway is geometrically impossible, fix that — and the fix changes the light on the arm, so the hand needs doing again. Then they grade the image, and the graded version reveals a seam around the hand that was invisible before.

Every one of those is a repeat of work already done. An order exists that prevents it, and it comes from one principle: **fix the things that change the most pixels first.**

The order

1. Structure. Anything that is geometrically wrong across a large area — a wall that does not meet the floor, a table with an impossible edge, a limb emerging from the wrong place. These need large masks and a high denoise, so they overwrite everything nearby.

2. Large removals and additions. Taking out an object, adding one. Same reason: large area, large change.

3. Faces and hands. The expensive, attention-demanding work. It comes after structure because a structural fix next to a face will disturb it.

4. Edges and boundaries. Where two things meet: hair against background, a product against a surface, a sleeve against an arm. These are small masks at moderate denoise.

5. Small objects and details. Jewellery, buttons, cutlery, the stems of glasses, wristwatches.

6. Global grade and grain. Contrast, colour, vignette, grain, sharpening. Always last, and always on a flattened copy, because grain applied before a repair means the repaired patch has no grain and shows.

Figure 5.2

The order that stops you doing the same repair twice

- **First**
  Structure. Anything geometrically wrong across a large area — a wall that does not meet the floor, a limb from the wrong place. Large masks, high denoise.
- **Second**
  Large removals and additions. Same reason: large area, large change, everything nearby overwritten.
- **Third**
  Faces and hands. The attention-demanding work, after structure, because a structural fix beside a face disturbs it.
- **Fourth**
  Edges and boundaries. Hair against background, a product against a surface. Small masks, moderate denoise.
- **Fifth**
  Small objects and details. Jewellery, buttons, cutlery, the stems of glasses.
- **Last**
  Global grade and grain, on a flattened copy. Grain applied before a repair leaves the repaired patch with none, and it shows.

The principle underneath is one line: fix the things that change the most pixels first. Every repair done out of order is a repeat of work already done.

The estimate, written down

Before starting, list the faults with minutes against each — the keeper checklist ended with this and it matters more here.

```
seed 418, base render
  1 doorway geometry, upper right      structure    8
  2 remove stray object on table       removal      4
  3 both hands                         hands       14
  4 face, slight asymmetry at 3/4      face         6
  5 bangle merging into wrist          edge         4
  6 chilli stems, foreground           detail       5
  7 grade, grain                       global       6
                                                   47
```

Forty-seven minutes is a normal figure for a hero image and a useful one to know, because it is what you are quoting for. If the list exceeds ninety minutes, go back to the contact sheet — a better base render is a minute away and will cost half as much to finish.

Working non-destructively

Every repair happens on a copy. The practical arrangement:

```
gen/v01/00418-base.png          never touched
work/hero.xcf                   layered file, repairs as layers
deliver/hero-v03.jpg            flattened output
```

In GIMP, Krita or Photopea, each inpainted region comes back as its own layer with a mask. That way a repair you later dislike is one layer to delete, rather than a reason to start again. It also means that when the client asks on Thursday whether the earlier hand was better, you can show them in four seconds.

The habit that costs people whole afternoons is inpainting in the generator, saving over the file, inpainting again, saving over it. Six passes later something is wrong and there is no way back to pass three.

Two failures specific to repair

The patch that does not belong. An inpainted region regenerated at full resolution often comes back sharper, more contrasty and slightly differently coloured than its surroundings, because it was denoised with a different

amount of context. It looks pasted. The fix is a low-denoise pass over the *joint* afterwards — mask a ring around the repaired area at 0.25 and let the model reconcile it.

Drift across many repairs. Twelve inpainting passes each shift the local colour slightly, and after twelve the image has a subtly uneven quality that no single region explains. This is why the grade goes last: a single global correction over an image with accumulated local drift fixes far less than people expect. Keep the number of passes down, use larger masks rather than many small ones, and check the whole image at 50% every few repairs rather than staying zoomed in.

When to stop

There is a point where repair stops improving the picture and starts sanding it flat. The signal is that you are fixing things nobody would notice — a slightly odd fold in fabric, an unconvincing reflection in a background window at 400% zoom.

View at 100% and at the delivered size. If a fault is invisible at both, it is not a fault. Clients do not zoom to 400%, and an image that has been repaired into perfect smoothness has usually lost the incidental irregularity that made it look like a photograph in the first place.

BEFORE YOU MOVE ON

Why must global grain and grading be applied after all masked repairs rather than before?

A. Because grading before repair shifts the colours the inpainting model matches against, so every repaired patch inherits the wrong base tone.

B. Because a repaired region regenerates without the grain that was applied earlier, so the patch shows as a smooth area.

C. Because grading is destructive and cannot be undone once the file has been flattened for the inpainting pass to run on it.

D. Because the inpainting model is trained on ungraded images and performs measurably worse when given a strongly graded region to complete.

The answer and why: p. 408

Lesson 44 · 8 min

The three repairs everybody needs, with settings

THE ONE THING TO KEEP

Hands, eyes and teeth fail because they are small, high-variance structures rendered at too few pixels, so the fix is always to give the region more pixels rather than to describe it better.

One cause, three symptoms

Hands have twenty-seven bones and appear in training data at every possible angle, occluded, in motion, half out of frame. Eyes must be symmetrical and matched to each other across a gap. Teeth are a repeating structure where any irregularity reads as wrong.

All three fail for the same underlying reason: in a 1024-pixel frame, a hand might occupy 60 pixels, which in latent space is about 8 × 8 — roughly sixty values to encode a structure that needs far more. The model is not being careless; it does not have the resolution.

Which tells you the fix. **Give the region its own full resolution.** Everything below is a variation on that.

The setting that does it

Every interface has a version of it. A1111 and Forge call it "Inpaint at full resolution" with a padding value; ComfyUI does it with a crop-and-stitch workflow; InvokeAI does it on the canvas by default.

What it does: crops the masked region plus some padding, scales that crop *up* to the model's native resolution, denoises it there, scales it back down and stitches it in. Your 60-pixel hand is generated as a 1024-pixel hand and then reduced, which is why it comes back with correct structure.

Without this setting, inpainting a hand is the model trying again at 60 pixels, and it will fail again.

Hands

```
mask:     the whole hand plus the wrist, generously
padding:  32-64 px
denoise:  0.45 - 0.6
prompt:   a relaxed open hand, five fingers, natural pose
steps:    same as base
```

Notes that matter:

- **Mask generously.** A tight mask around bad fingers forces the model to connect to the existing wrong geometry. Include the wrist and some background.
- **Prompt for the hand, not the scene.** During an inpaint the prompt should describe the region. Leaving the full scene prompt in place tends to introduce scene elements into the patch.

- **Simplify the pose.** If the hand is gripping something at an awkward angle, a relaxed or partly hidden hand often reads better and generates first time.
- **Generate four and pick.** Hands are high-variance; batch them rather than iterating one at a time.

If four attempts all fail, the pose is outside what the model does. Composite a photographed hand, or crop it out.

Eyes

```
mask:     both eyes together, never one at a time
denoise:  0.35 - 0.5
prompt:   clear symmetrical eyes, natural catchlight, looking at
camera
```

Masking both together is the whole technique. Eyes have to agree with each other — direction of gaze, size, catchlight position — and repairing one in isolation guarantees they will not. It is the single most common eye-repair mistake.

Keep denoise lowish. Eyes carry identity, and above about 0.55 you are generating a different person's eyes.

Teeth and mouths

```
mask:     the mouth and a margin of lip
denoise:  0.3 - 0.45
prompt:   natural teeth, slight smile, soft lip edge
```

Teeth want *less* denoise than you would expect, because the failure is usually irregularity rather than absence. High denoise produces the other classic failure: a perfect, unnaturally even set that looks like dentures. Some asymmetry is what makes a mouth read as real.

Automatic detection tools

ADetailer and its equivalents automate this: they detect faces, hands or eyes with a small detector, mask each one, and run a full-resolution inpaint pass on it. It is free, it plugs into most interfaces, and for a batch of twenty images it is the difference between an afternoon and five minutes.

Two cautions. It applies the same denoise to every detection, so a face that was already fine gets regenerated and may change. And on a set, that regeneration can shift a character's identity slightly between frames, which is exactly what the consistency block is trying to prevent. Use it for volume, check every result, and turn it off when identity matters.

The honest ceiling

Some hands do not come back. Two hands interlocked, fingers gripping a complex object, a hand pressed against a face — these are configurations where even a full-resolution attempt fails repeatedly, because the training data is thin there and the structure is genuinely ambiguous.

At that point the professional answer is the unglamorous one: photograph your own hand in that position with a phone, and composite it. It takes ten minutes, it works every time, and it is what retouchers have done for decades. Persisting with a twelfth inpaint attempt is a decision to spend an hour on something a photograph solves, and recognising that moment is worth more than any setting in this lesson.

BEFORE YOU MOVE ON

Why does "inpaint at full resolution" fix hands when ordinary inpainting at the same denoise does not?

A. Because the cropped region is scaled up to the model's native size before denoising, so the hand is generated with far more pixels.

B. Because cropping to the masked region removes the surrounding context that was biasing the model toward reproducing the same incorrect hand structure.

C. Because the setting increases the effective step count over the masked area, allowing the finer structure of the fingers to converge properly.

D. Because a smaller region occupies less of the model's attention budget, so the fingers receive proportionally more of the conditioning signal than they did in the full frame.

The answer and why: p. 409

Lesson 45 · 8 min

Type is set, not generated

THE ONE THING TO KEEP

Generated text fails at unusual words, small sizes and long strings because a diffusion model renders letterforms as visual texture rather than assembling glyphs, so any text that must be correct is set in a type tool over a clean plate.

What the model is doing with letters

A diffusion model has no glyphs and no notion of spelling. It has learned what text *looks like* in photographs: the visual texture of a shop sign, the arrangement of marks on a book spine, the rhythm of a paragraph. When it produces text it is producing that texture, and sometimes the texture happens to spell something.

Newer models do markedly better, because they were trained with better captions and stronger text encoders, and short common words in large clear type now land often. But the mechanism has not changed, and that tells you exactly where it will still fail:

Figure 5.3

What is still wrong with generated letters, and what type gives you

Where a model still fails

- Long strings — error accumulates per character
- Unusual words, brand names, proper nouns
- Small text, for the same reason hands fail
- Devanagari, Arabic, Tamil and other joined scripts
- A specific typeface, weight, tracking and alignment

What setting the type gives you

- Correct spelling at any length
- The brand's actual typeface
- Copy a client can change without regenerating
- Kerning, tracking and alignment you control
- Vector, so it stays sharp at any size

This separation is not a workaround waiting for better models. Type and image are edited on different schedules, in different tools, by different people, which is why no newspaper has ever photographed its headlines.

- **Long strings.** Error accumulates per character.
- **Unusual words.** Brand names, proper nouns, anything not common in training text.
- **Small text.** Not enough pixels, the same problem as hands.

- **Non-Latin scripts.** Devanagari, Arabic, Chinese, Tamil — thinner training coverage, and in scripts with contextual joining or conjuncts the failures are severe and frequently nonsensical.
- **Exact typography.** A specific typeface, weight, tracking and alignment.

The professional answer

You do not generate text. You generate a clean plate — the image with an empty area where the text goes — and you set the type in a type tool.

This is not a compromise. It is how every poster, advertisement and book cover you have ever seen was made. The photograph and the typography were always separate operations, and nobody would dream of asking a camera to render a headline.

The free tools are complete:

- **Inkscape** — vector, full typographic control, exports print-ready PDF. The right tool for a poster or any layout.
- **GIMP** or **Krita** — raster type over an image. Fine for social assets.
- **Scribus** — page layout, for anything multi-page or going to a commercial printer.
- **Photopea** — in a browser, reads PSD, useful on a locked-down machine.

The workflow

1. Generate with the text area planned as quiet space, as the brief block set out. Prompt for `plain wall`, `overcast sky`, `dark background` in that region.
2. Do not prompt for text at all. Asking for a sign and then covering it wastes effort and often produces garbled marks that show through.
3. Open the plate in your type tool.
4. Set the type. Check contrast against the background — 4.5:1 minimum for body-sized text is the accessibility threshold and a good rule generally.
5. If the background is busy, add a scrim: a soft dark gradient or a semi-transparent panel behind the type. Cheap, and it works.

When you want the text to be in the scene

Sometimes the type has to look like it belongs to the world — a shop sign in perspective, a label on a bottle, a book cover on a table.

The method is the same, with one more step:

1. Generate the scene with a blank sign, blank label or plain spine.
2. Set the type flat in your tool.
3. Transform it to match the surface — Perspective transform in GIMP, or a free-transform in Photopea.
4. Match its behaviour to the scene: reduce opacity slightly, apply the same blur if the surface is out of focus, add the same grain, and multiply or overlay it so the surface texture shows through.

That fourth step is what separates a convincing result from an obvious paste-on. A perfectly crisp, perfectly opaque piece of type on a slightly soft, grainy photograph announces itself immediately.

When generated text is acceptable

Two cases, and only two.

Decorative illegibility. Background signage in a street scene that nobody will read. Here garbled text is fine — except when it accidentally spells something, which is a real risk and belongs on the forbidden-list check.

A rough concept for internal review. Showing a client where a headline will sit, clearly marked as placeholder.

Anything a viewer will actually read gets set properly.

The limitation that will not go away soon

Even where a model renders a short word correctly, it has no typographic control. You cannot specify the typeface, the weight, the tracking, the alignment or the exact size — and brand work is defined by exactly those things. A brand's typeface is usually its second-most-protected asset after the logo.

So the separation of image and type is not a temporary workaround to be abandoned when models improve. It is the correct structure for the work, because type and image are edited on different schedules, in different tools, by different people, and a headline needs to be changeable without regenerating a photograph. Even if a model rendered perfect type tomorrow, you would still set it separately, for the same reason a newspaper does not photograph its headlines.

BEFORE YOU MOVE ON

Why is setting type in a separate tool the correct approach rather than a temporary workaround for current model limitations?

A. Because diffusion models will always render letterforms as texture, so generated text can never be reliable at any quality level.

B. Because vector type scales to any output size without resampling, whereas generated raster letterforms degrade once they are enlarged to print resolution.

C. Because typeface, weight, tracking and alignment cannot be expressed in a prompt, and a headline must be editable without regenerating the picture.

D. Because generated text may accidentally reproduce a trademarked wordmark or an unfortunate string, creating a rights problem that deliberately set type avoids entirely.

The answer and why: p. 409

Lesson 46 · 8 min

Extending a frame when the crop changes

THE ONE THING TO KEEP

Outpainting extends an image in overlapping steps of a few hundred pixels, and the overlap is what lets the model match light, perspective and texture rather than inventing a discontinuous new scene.

The call that makes you need this

The image is approved at 4:5. Marketing now needs a 16:9 version for the website banner. Cropping a 4:5 frame to 16:9 removes most of the picture, and the subject goes with it.

Outpainting adds canvas and fills it with content that continues the image. Done properly it is invisible. Done in one big jump it produces a seam, a change of light, and a background that does not agree with itself.

The method

Extend in steps of 128–256 pixels, not in one leap.

1. Enlarge the canvas by 256 pixels on the side you need.
2. Mask the new empty area **plus 64–128 pixels of the existing image** — the overlap is the important part.
3. Inpaint with denoise 0.8–1.0 over the new area, using a prompt describing what continues there, not the original scene prompt.
4. Repeat until you reach the size you need.

Why steps: the model sees only its context window. A 512-pixel extension in one pass means most of the new region has no nearby real pixels to match against, so it invents freely — different light, different perspective, sometimes a different room. A 256-pixel step keeps every new pixel close to something real.

Why the overlap: without it, the model is generating a strip that must meet the original exactly at a hard boundary, and it will not. With 100 pixels of overlap it is *continuing* an image rather than abutting one, and the transition happens inside the regenerated area where nobody can see it.

Prompting an extension

Describe what should be there, not what the image is of.

- Extending upward from a courtyard scene: `whitewashed wall continuing upward, overcast sky above the roofline` — not the full original prompt.
- Extending sideways from a portrait: `plain studio background, soft falloff` — not the description of the person, which will get you a second person.

That second-person failure is the most common outpainting mistake. The prompt still describes a woman, the model has empty canvas, and so it puts a woman there.

What breaks

Perspective drift. Each step can shift the vanishing point slightly, and after four steps a straight wall curves. Check long straight lines after every step rather than at the end. If drift starts, a control — canny on the existing edges, or a depth map — holds the geometry.

Light drift. The new area gradually becomes brighter or flatter. Same cause, same remedy: check each step, and correct the grade as you go rather than at the end.

Repetition. Extending a patterned surface produces obvious repeats, because the model continues what it sees. Vary the prompt slightly on each step, or break it up with a repair pass afterwards.

The tell at the join. Look specifically for a faint vertical or horizontal line where the seam was, visible as a slight change in noise or contrast. A low-denoise pass at 0.2 over a band spanning the join clears it.

Doing it in one operation

Some tools handle this well in a single step — Photoshop's generative expand, InvokeAI's canvas, and several hosted services all extend a frame cleanly in one action. If you have one and the extension is modest, use it.

The manual method still matters for two reasons. Large extensions — doubling a frame's width — fail in one shot regardless of tool, because the problem is context rather than implementation. And the manual method gives you control over the prompt at each step, which is how you decide what is in the new space rather than accepting what the model thinks belongs there.

The better answer, when there is time

Outpainting is a rescue. The better outcome is not needing it, and that is the brief block's job: ask at the start which ratios are required, and either generate wider than any of them or generate each ratio separately.

There is also a real ceiling. Outpainting extends a *scene*; it cannot extend a *composition*. If the original frame worked because the subject was placed against the edge, adding 400 pixels of room changes that relationship, and the wider version can be technically seamless and compositionally worse. Look at the extended image as a new picture and judge it as one, rather than only checking whether the join shows. Sometimes the honest answer to the banner request is a fresh generation at 16:9 using the approved frame as a reference, and it takes less time than four careful extension steps.

BEFORE YOU MOVE ON

Why does outpainting in 256-pixel steps with overlap work better than one 512-pixel extension?

A. Because a smaller masked area allows a higher denoise value to be used safely, which produces more convincing new content in the extended region.

B. Because each new pixel stays close to real pixels it can match light and perspective against, and the overlap hides the join inside regenerated area.

C. Because the model processes the image in fixed tiles, so an extension larger than one tile is composited from separate passes that cannot agree with each other.

D. Because repeated smaller passes allow the prompt to be adjusted between steps, which is the main reason the incremental approach produces better results.

The answer and why: p. 410

Lesson 47 · 8 min

Four ways to remove an object, and when each is right

THE ONE THING TO KEEP

Clone, heal, content-aware fill and inpainting fail in different ways, so the choice depends on whether the area behind the object is a texture, a structure or a scene that has to be invented.

The tools, and what each actually does

Clone stamp copies pixels from one place to another exactly. You choose the source. Total control, no invention, and it repeats visibly if you are careless. Available in GIMP, Krita and Photopea.

Healing brush copies texture from a source but blends the *colour and luminance* of the destination. This is why it works where clone fails: a spot on a gradient healed from a flat area takes the texture of the source and the tone of the target, so it disappears. It fails when you cross a boundary, smearing two regions together — which is why healing near an edge produces the characteristic muddy blur.

Content-aware fill (GIMP's Resynthesizer, Photopea's equivalent) analyses the surrounding area and synthesises a patch from it algorithmically. Good at texture — grass, gravel, sky, fabric. Poor at structure, because it has no concept of a straight line and will happily curve a window frame.

Inpainting with a diffusion model regenerates the region from the prompt and the context. It is the only one of the four that can *invent* something that was never in the image — a continuation of a wall that is entirely hidden, a piece of floor with a pattern that has to be plausible rather than copied.

The decision

Look at what is behind the thing you are removing.

- **Uniform texture** — sky, wall, grass, water. Content-aware fill. Two seconds and it is done.
- **A gradient with no structure** — a studio backdrop, a soft sky. Healing brush, or clone with a soft edge.
- **Regular structure** — tiles, brickwork, panelling, a railing. Clone stamp, sampling from a matching part of the same structure. This is where the algorithms fail and patience wins.
- **Complex scene content** — the object was in front of a shelf of things, a crowd, a detailed background. Inpainting.
- **A large object occupying a third of the frame** — inpainting with a generous mask, then a repair pass on the joint.

Most real removals use two: inpaint the bulk, then clone-stamp the structural edge the model got wrong.

Inpainting a removal properly

The instinct is to mask the object tightly and prompt for what should be there. Two adjustments make it work far better.

Mask beyond the object. Include its shadow, its reflection, and 20–30 pixels of surroundings. An object's shadow is the thing people forget, and a removed object leaving its shadow is the most conspicuous possible failure.

Prompt for the background alone, not for absence. `empty wooden table, soft shadow` works. `no cup` does not — negation is unreliable, and naming the cup in any form raises the chance of getting one.

Denoise should be high, 0.8–1.0, because you genuinely want new content rather than a modification of what is there.

The ghost

The characteristic failure of any removal is a ghost: a faint outline, a slight difference in tone, a shape you can see when you look for it. Three causes, three fixes.

Tonal mismatch — the patch is slightly lighter or darker. Correct with a curves adjustment confined to the patch, not to the whole image.

Texture mismatch — the patch is smoother than its surroundings because grain was lost in regeneration. Add matching noise, or run a low-denoise pass at 0.2 over the area.

Edge residue — a rim of the original object's colour survives at the mask boundary. This is the mask being too tight. Mask wider and redo it; feathering alone will not clear a contaminated edge.

Check by exaggerating: duplicate the layer, apply a strong Curves contrast, and look. Ghosts that are invisible normally are obvious under high contrast, and this thirty-second test catches almost all of them before a client does.

What removal costs you

An honest limitation. Removing an object means the information behind it does not exist, and everything you put there is fabricated. On an artistic image that is fine. On an image that documents something real, it is an alteration of a record, and the difference is not technical.

There is a second, subtler cost. A scene with an object removed often composes worse than it did, because the object was doing something — balancing the frame, filling a corner, giving the eye somewhere to rest. People remove a distraction and find the picture has gone slack.

So look at the result as a composition, not just as a successful removal. Occasionally the right answer is to replace the object with something else rather than leaving a gap, and occasionally it is to leave it alone.

BEFORE YOU MOVE ON

An object is removed from a gravel path but leaves a faint smooth patch visible when the image is examined. What is the cause and remedy?

A. The mask was feathered too heavily, blending the original object's tone into the surrounding gravel across a wide transition band.

B. The removal was performed with content-aware fill, which cannot synthesise irregular natural texture and should have been an inpainting pass.

C. Texture was lost in regeneration, so the patch lacks the grain of its surroundings; add matching noise or run a low-denoise pass over it.

D. The denoise value was set too high, which regenerated the region without reference to the surrounding texture and produced an unrelated smooth surface.

The answer and why: p. 410

Lesson 48 · 9 min

The real product in a generated world

THE ONE THING TO KEEP

A photographed product composited into a generated scene is convincing only when perspective, light direction, contact shadow, reflection and grain all agree, and the contact shadow is the one that gives it away.

The workflow that does most commercial work

The brief block established it: generate the world, photograph the claim. This lesson is the execution, and it is the single most commercially useful technique in this course.

The result is a product shot that could not have been photographed — a bottle on a mountain road at dawn, a jar in a 1950s kitchen — in which the product is genuinely, exactly, the client's product.

Figure 5.4

A real product in a generated world, in five moves

- **One**
  Generate the environment with the product's place left empty. Prompt for it, or block it out in Blender. Decide the light direction now and write it down.
- **Two**
  Photograph the product in light that matches: same direction, same hardness, same camera height, roughly the same focal length.
- **Three**
  Cut it out. Background removal with rembg or Foreground Select, then a one-pixel refinement of the edge by hand.
- **Four**
  Place it, then add the contact shadow, the cast shadow, the reflection and bounce, and grain that matches the scene.
- **Check**
  Zoom out to twenty-five per cent. Composites fail at small size before they fail at large size, because the eye reads integration before edges.

The contact shadow is the one people forget and the one that gives it away. Perspective and light direction are decided at step two and are effectively unfixable afterwards.

Step one: reserve the space

Generate the environment with a place for the product. Two ways:

- **Prompt for it**: `an empty wooden table in the foreground, morning light from the left, shallow depth of field`. Leave the space empty rather than generating a similar object and covering it, because a generated object's shadow will show around your real one.
- **Blockout it**: a cylinder in Blender at the product's position and size, depth-conditioned. This gives you correct perspective and, usefully, tells you exactly where the product must sit.

Decide the light direction now and write it down. Everything downstream depends on it.

Step two: photograph the product to match

The photograph must match the scene's light, and this is where people fail. If the generated scene is lit hard from the left, the product must be lit hard from the left.

With a window and two sheets of paper:

1. Product side-on to the window, matching the scene's direction.
2. White paper on the shadow side if the scene's shadows are soft; remove it, or use black card, if they are hard.
3. Plain background — white paper, or black cloth.
4. **Match the camera height and angle to the scene.** If the generated table is seen from slightly above, photograph the product from slightly above. Mismatched viewpoint is unfixable in an editor.
5. Match the focal length roughly. A scene prompted at 35mm wants a product shot from a similar distance, not a phone held at arm's length in portrait mode.
6. Thirty frames, keep two.

Step three: cut it out

Background removal, then edge work. Free tools handle this well: GIMP's Foreground Select, Krita, Photopea, or `rembg`, which is a small free model that does a good first pass in one command.

Then clean the edge by hand. A 1-pixel refinement around a bottle is what separates a composite from an obvious paste. If the product has a transparent or reflective part — glass, chrome — the background will show through it and the cutout must preserve that, which is hand work and takes time.

Step four: place, and then do the four things that matter

Placement is the easy part. These four are what make it believable.

Contact shadow. The single most important element, and the one people forget. Where an object meets a surface there is a dark, tight, soft-edged shadow. Without it the product floats, and every viewer feels it even if they cannot name it. Paint it on a new layer in Multiply mode with a soft black brush at low opacity, hard and dark directly under the base, fading outward within about 10% of the object′s width.

Cast shadow. The longer shadow thrown across the surface, in the scene′s light direction, softening and lightening as it recedes. Match its angle to every other shadow in the frame.

Reflection and bounce. On a glossy surface, a faint flipped copy at low opacity. From a coloured surface, a hint of that colour on the product′s lower edge.

Grain and optics. The generated scene has grain, a particular sharpness and probably some depth-of-field falloff. Your photograph does not. Match them: add the same grain, blur the product slightly if it sits in a plane that should be soft, and match the colour temperature.

The check

Zoom out to 25%, then look at the small version. Composites fail at small size before they fail at large size, because the eye reads overall integration rather than edges.

Then the three questions: does the shadow point the same way as every other shadow; is the product the size it would really be next to the objects around it; does it have the same grain and sharpness as its surroundings. Two of those

will be right and one will not.

Where this is the honest choice

This is not a trick to make people think a photograph was taken. It is the standard composite workflow of commercial photography, done since long before generative models, with a generated backdrop instead of a painted or photographed one.

Two lines are worth keeping. The product must be the actual product, unaltered in a way that misrepresents it — no slimmed bottle, no enhanced colour that the item does not have. And the environment must not imply a claim that is false: a food product photographed in a generated restaurant that does not exist, presented as the restaurant serving it, is a different matter from a jar on a generated marble surface.

Where exactly that line sits depends on the market and on advertising rules that differ by country. The business block returns to it. The practical version: if the environment would be understood as a stage set in a photograph, it is fine generated. If it would be understood as documentary evidence, it is not.

BEFORE YOU MOVE ON

Which element most reliably gives away a composited product as not belonging to the scene?

A. The absence of a tight contact shadow where the object meets the surface, which makes it appear to float.

B. A mismatch in resolution, because a photographed product carries far more fine detail than the generated background it is placed against.

C. An imperfect cutout edge, where remnants of the original photographic background remain visible as a fringe around the object's outline.

D. A difference in colour temperature between the photographed product and the surrounding generated environment, which the eye detects immediately at any size.

The answer and why: p. 411

Lesson 49 · 8 min

Making two sources agree

THE ONE THING TO KEEP

Every image carries a signature of noise, sharpness, depth of field and chromatic behaviour, and a composite reads as fake when two sources disagree on that signature rather than on content.

What the eye is actually detecting

Show someone a composite and they will say it "looks off" without being able to say why. They are almost never responding to the content. They are responding to two images having different *physical* characteristics, which the visual system reads as two surfaces rather than one.

Five characteristics, all matchable.

Grain and noise

Every photograph has noise — sensor noise, film grain — with a particular size and colour. Generated images have their own, usually finer and more even, and often almost none after upscaling.

The fix: flatten the composite and add a single grain layer over everything. In GIMP that is a new layer filled with 50% grey, `Filters > Noise > RGB Noise` at a low value, set to Overlay, opacity 10–25%.

Doing it over the whole image rather than per element is the important part. A unified grain layer makes disparate sources belong to one photograph, and it is the cheapest, most effective integration step there is. This alone rescues most composites.

Sharpness and depth of field

If the generated scene has a shallow depth of field and your product is pin-sharp, the product will not sit in the space. Work out which plane the object occupies and blur it to match, using a graduated blur if it is large enough to span planes.

The opposite happens too: an upscaled generated background can be unnaturally crisp everywhere, sharper than any real lens, and a normally-photographed product looks soft against it. Then you blur the background slightly rather than sharpening the product.

Chromatic aberration

Real lenses split colour slightly at high-contrast edges — a faint red-cyan or blue-yellow fringe, strongest at the frame's corners. Generated images often have it, inherited from training data. A cut-out product will not.

If the scene has visible fringing, add a trace to the product: offset the red and blue channels by a pixel. If it does not, remove any the product has. Either way, both sources must agree, and this is a detail that separates careful work from adequate work.

Vignette and falloff

Real lenses darken toward the corners. Add one vignette over the flattened composite rather than carrying two.

Colour temperature and black level

Two sources will have different whites and different blacks. Match the black point first — one Curves adjustment lifting or lowering the darkest values — then the white balance. People do it the other way round and then cannot get the shadows to agree, because a mismatched black level looks like a colour cast and is not one.

The order

1. Place and scale the elements.
2. Match black point, then white balance, per element.
3. Match depth of field and sharpness, per element.
4. Match chromatic aberration, per element.
5. Add contact shadows and cast shadows.
6. **Flatten a copy.**
7. Apply grain, vignette and final grade to the whole thing.

Steps six and seven are the ones that do most of the work and the ones most often skipped. Global treatment over a flattened image is what makes separate sources become one photograph.

Free tools, all of it

GIMP, Krita and Photopea between them do every step. GIMP has Curves, RGB Noise, Gaussian blur, channel offsets for aberration, and a gradient tool for vignettes. If you want a more film-like grain, `Filters > Noise > HSV Noise` is usually better than RGB noise because it varies hue slightly, which is closer to how film behaves.

For a graded look, GIMP supports colour lookup tables, and free cinematic LUTs are widely available — one applied at reduced opacity over the flattened composite unifies everything and gives the set a house look for free.

What this cannot fix

Matching signatures fixes *integration*. It does not fix *geometry*. If the product was photographed from eye level and the scene is seen from above, no amount of grain matching will make it sit in the space — the viewer reads perspective before texture, and a perspective mismatch is visible at thumbnail size while a grain mismatch is not.

So perspective and light direction are decided when you photograph, and are effectively unfixable afterwards. Grain, sharpness, aberration and grade are fixable at any point. Knowing which category a problem falls into tells you

whether to open the editor or to reshoot — and reshooting a product against a window takes ten minutes, which is less than an hour of failing to correct perspective in software.

BEFORE YOU MOVE ON

Why should grain be applied to a flattened composite rather than to each element separately?

A. Because per-element grain would require matching the noise characteristics of each source individually, which is difficult to judge by eye at working zoom levels.

B. Because applying grain to a layer with transparency produces artefacts at the alpha edges that are visible once the composite is flattened.

C. Because a single grain layer over everything is what makes separate sources read as one photograph.

D. Because grain applied before flattening is resampled when layers are transformed, which softens the noise and reduces its integrating effect.

The answer and why: p. 411

Lesson 50 · 8 min

The fault the client sees on a big screen

THE ONE THING TO KEEP

A composite or repair fails at its boundary, and the three specific faults — colour contamination, a bright or dark halo, and over-feathering — each have a distinct cause and a distinct fix.

Why edges matter more than anything else

An edge is where two things meet, and it is where the eye goes. A perfect cutout with a one-pixel white fringe reads as a cutout instantly. A beautifully repaired hand with a visible ring around the repair reads as a repair.

And edges are the fault that survives review. You look at the image at 66% on a laptop; the client opens it at 100% on a large monitor in a meeting. The halo you could not see is a line they can.

Fault one: colour contamination

Cut a dark-haired subject out of a bright green background and the hair edge retains green pixels. Place them on a warm background and there is a green rim.

The cause is real: at the boundary, individual pixels genuinely contain a mixture of subject and background, because that is what happened at the sensor. A binary mask cannot separate what is physically blended.

Fixes, in order of preference:

- **Decontaminate colours** — GIMP's Foreground Select offers this, and it estimates and removes the background contribution from edge pixels. Use it whenever it is available.
- **Shrink the selection by 1 pixel** before cutting. Crude, loses a little of the subject, works.

- **Paint it out** on a duplicated layer clipped to the subject, using a soft brush in Colour mode sampling from the adjacent correct hue.
- **Reshoot against a neutral background** if the subject is yours to reshoot. Grey or white contaminates far less visibly than green or blue.

Fault two: the halo

A bright or dark line following the boundary. Two distinct causes, needing different fixes.

From sharpening. Unsharp masking increases contrast at edges, which by construction creates a light band on one side and a dark band on the other. Overdone, it is visible as a halo. The fix is less sharpening, and sharpening last on the flattened file with a small radius — 0.5 to 1 pixel for screen output.

From upscaling. Some upscalers produce ringing at high-contrast boundaries. The fix is a different upscaler, or masking the artefacted areas and repairing them.

Distinguish them by looking at whether the halo appears on all high-contrast edges in the image, which indicates sharpening or upscaling, or only around your composited element, which indicates a masking problem.

Fault three: over-feathering

The opposite failure. Feather a mask by 20 pixels to avoid a hard edge and you get a soft, blurry boundary that looks like a badly cut sticker.

Correct feathering depends on what the edge is:

- **A hard object edge** — a bottle, a box, architecture: 0.5 to 1 pixel. Almost none.
- **A soft edge** — hair, fur, fabric in motion: 2 to 4 pixels, plus proper edge work.
- **A repaired region blending into similar texture**: 8 to 20 pixels is fine, because there is no real boundary to preserve.

The common error is applying a single feather value everywhere. A cut-out product needs a sharp edge and its shadow needs a very soft one; they are different masks.

Checking properly

Three tests, ninety seconds:

1. **Zoom to 200%** and trace the entire boundary. Slow, and it is the only way to be sure.
2. **Contrast test.** Duplicate the flattened image, apply extreme Curves contrast, and look. Halos and contamination jump out.
3. **Background swap.** Put a mid-grey layer behind a cutout, then a saturated one. Edge faults invisible against the intended background are obvious against a contrasting one.

The one that is not a mistake

Some edge softness is correct and should be left alone. A real photograph has softness at edges from depth of field, motion and the lens itself. A subject cut out with a surgically perfect edge and placed into a scene with natural softness will look *too* sharp, and that reads as fake just as strongly as a halo.

So match the edge to its context. If the scene's other edges at that depth are slightly soft, yours should be too. This is the same principle as the previous lesson — the two sources must agree — applied at the boundary rather than across the surface.

It also means an edge can be technically perfect and still wrong for the image, which is the sort of thing that separates people who have done this for a while from people who are following a checklist. The checklist gets you to competent. Looking at what the rest of the photograph is doing gets you past it.

BEFORE YOU MOVE ON

A cut-out subject shows a faint light line along its entire boundary, and the same line appears on other high-contrast edges elsewhere in the image. What is the cause?

A. Colour contamination from the original background surviving in the partially blended edge pixels of the cutout.

B. The mask was feathered too little, leaving a hard boundary that the eye reads as a bright line against the darker surrounding content.

C. Sharpening or upscaling applied to the whole image, which raises contrast at every boundary and produces a light band on one side.

D. The composited layer's blend mode is lightening the pixels beneath it around the boundary, where the layer's alpha value falls between fully opaque and fully transparent.

The answer and why: p. 412

Lesson 51 · 8 min

Getting to poster size without a second horizon

THE ONE THING TO KEEP

A large final image is produced by generating at native size and enlarging in a separate pass, and where extra generated detail is needed the canvas is processed as overlapping tiles with a low denoise so no tile can invent a new scene.

Two different problems

"I need a bigger image" hides two requirements that need different answers.

More pixels, same detail. A 1024 render that must print at A3. Nothing needs to be added; the existing picture needs to be larger. This is upscaling, and the finishing block covers what upscalers add and invent.

More detail at large size. A poster where the fabric texture, the foliage and the distant faces all need to hold up at 60 cm. This needs new detail generated, and that is what tiled diffusion is for.

Confusing them wastes time. If the image will be seen from two metres, it needs pixels, not detail.

Why you cannot just generate large

Covered in the brief block and worth restating: past its trained pixel count, a model composes badly — two horizons, duplicated subjects, a scene assembled from overlapping attempts. Memory is not the limit; composition is.

How tiled diffusion works

Ultimate SD Upscale, tiled diffusion and their equivalents all do the same thing:

1. Upscale the image conventionally to the target size — say 1024 to 4096 with an ESRGAN-family model.
2. Divide it into overlapping tiles, typically 512 or 768 with 64 pixels of overlap.
3. Run img2img on each tile at **low denoise**, 0.2–0.35, with a prompt.
4. Blend the overlaps.

The low denoise is what makes it safe. Each tile is being refined, not re-imagined, so no tile can decide there should be a second horizon in it. Push denoise above about 0.4 and tiles start inventing content, which produces the characteristic failure: a wall that becomes brickwork in one tile and plaster in the next.

Settings that matter

Denoise 0.2–0.35. The single most important value. Above 0.4 expect inconsistency.

Overlap 64 pixels minimum. Less and seams show. More is safer and slower.

A generic prompt. This is the setting people get wrong. Each tile gets the same prompt, and a tile containing only sky given a prompt about a woman in a sari will try to put her there. Use a short prompt describing the *medium and finish* — `documentary photograph, fine grain, natural light` — not the subject.

Seam fix if available, which runs an extra pass over the tile boundaries.

When not to use it

Faces. A face spanning two tiles gets two independent treatments and the result is subtly wrong. Repair faces at full resolution *before* tiling, and if a face still degrades, mask it out of the tiled pass.

Text, logos, product labels. Same problem, worse consequences. Anything that must remain exactly itself should be composited in after tiling, not put through it.

Large flat areas. Tiles introduce variation into flat surfaces, so a smooth studio backdrop acquires a faint patchwork. Mask it out or accept a plain upscale there.

The free path

All of this is free. Ultimate SD Upscale is an extension for A1111 and Forge; ComfyUI has tiled nodes in its core and its common node packs. The upscaler models themselves — ESRGAN, RealESRGAN, and the various community variants — are small files, run on modest hardware, and are free.

For the non-generative half, a plain upscale, GIMP's Scale Image with LoHalo interpolation is decent, and the free ESRGAN command-line tools are better. Neither needs a GPU worth the name.

The limit, and the professional reality

Tiled diffusion adds *plausible* detail, not *correct* detail. At 4096 the fabric has weave, the wall has texture, the foliage has leaves — and none of it corresponds to anything. For a landscape or an atmospheric image that is exactly right. For a product where the client knows what the label says, it is a fabrication that gets more detailed the larger you print it.

Which produces a rule worth holding: **the parts of the image that must be correct do not go through a generative upscale.** Composite them in at full resolution afterwards.

The other honest point is that large-format work rarely needs this at all. Viewing distance does most of the job. A roll-up banner read from two metres is fine at 150 dpi; a billboard at 30 metres is fine at 15. Before spending forty minutes on a tiled pass, calculate what the eye can actually resolve at the viewing distance — it is frequently less than the file you already have, and the whole exercise turns out to be unnecessary.

BEFORE YOU MOVE ON

Why must tiled diffusion run at a denoise value around 0.25 rather than 0.6?

A. Because higher denoise values increase the memory required per tile, which defeats the purpose of tiling a large canvas in the first place.

B. Because each tile is denoised independently, so at higher strengths a tile can invent content that contradicts its neighbours.

C. Because the overlap regions are blended after generation, and stronger denoising produces larger differences between the two versions of each overlapping strip.

D. Because tiled passes run after a conventional upscale, and the upscaled image already contains interpolation artefacts that a high denoise value would amplify across the frame.

The answer and why: p. 412

CASE STUDY · MODULE 5

A 4:5 frame, a 16:9 banner, and two hours

An illustrative case, built from documented patterns.

A one-person studio in Pune, delivering the launch assets for a hardware startup's first product page.

The hero image had been approved on Tuesday. It was 4:5, generated at 896 by 1152 and finished properly: the product photographed against a window and composited into a generated workbench scene, contact shadow painted in, grain matched, forty minutes of repair on the bench edges and a stray reflection. The client had signed it off in writing.

At four o'clock on Thursday, two hours before the page went live, the client's developer said the desktop hero slot was 16:9 and always had been. Nobody had asked the question at the start, and the spec — which Kiran had written — said *Hero: 1200 × 1500* and nothing else.

Cropping a 4:5 frame to 16:9 removes the top and bottom, which is where the product and the light source were. The composition dies. So the real options were two, and both had a clock on them.

Outpaint the sides in steps. Enlarge the canvas by 256 pixels, mask the new area plus about 100 pixels of the existing image, inpaint at high denoise with a prompt describing what continues there, repeat until the ratio is right. That is roughly four steps each side. Each step risks perspective drift on the workbench edge, light drift as the new area gradually brightens, and the characteristic second-subject failure if the prompt still describes the product.

Regenerate at 1344 × 768 using the approved frame as an img2img reference at low strength, then re-composite the photographed product, repaint the contact shadow, and redo the grade. A clean 16:9 composition, and about ninety minutes of work that has already been done once.

Kiran chose outpainting, for a reason that was correct at four o'clock: the approved frame was approved, and a regeneration is a new picture that the client has not seen and might not like at six. Signed-off work is worth protecting when the clock is the binding constraint.

The execution nearly went wrong in exactly the documented way. On the first attempt he extended 400 pixels in one leap with the original scene prompt still in the box. The model had empty canvas and a prompt describing a product on a workbench, so it put a second product on a second workbench, at a slightly different scale, with a light source that did not agree with the first. He deleted it, went back to 256-pixel steps, and rewrote the prompt for each step to describe only what continues there — *plain workshop wall continuing left, soft falloff* — rather than what the image is of.

After the third step on the left side he checked the long straight edge of the bench, as the lesson says to do at every step rather than at the end, and found it had begun to curve. He added a canny control on the existing edges for the fourth step and it held. On the right side he saw a faint vertical line at the second join, visible as a slight change in contrast, and cleared it with a low-denoise pass at 0.2 over a band spanning the seam.

The last thing he did, at twenty to six, was the thing that is easy to skip: he looked at the extended image as a new picture rather than checking whether the join showed. The original frame worked because the product sat close to the right edge with tension against it. With 400 more pixels of room on that side, the tension was gone and the product floated in the middle of a wide empty bench.

YOUR ANSWERS

1. Why did the first 400-pixel extension produce a second product, and what two changes fixed it?

2. Kiran protected an approved frame rather than regenerating. When would the other choice have been right?

3. The extension was technically seamless and compositionally worse. What did that cost, and how was it recovered?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 266.

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

CASE STUDY · MODULE 5

How it played out

He cropped back in. The final 16:9 was not the full extension — it used about two thirds of the new canvas on the left and almost none on the right, which put the product back near the right edge where the composition had wanted it. That took four minutes and was possible only because he had generated more room than he needed rather than exactly enough. The page went live at six with the 4:5 and the 16:9 from one approved frame. The next morning he added two lines to his spec template, before the deliverables list: *All required ratios, including any the development team knows about*, and *Generate at the widest required ratio; crop the others from it*. Three jobs later, that second line saved an afternoon on a brief that turned out to need a 9:16 story frame nobody had mentioned either.

The questions, discussed

1. Why did the first 400-pixel extension produce a second product, and what two changes fixed it?

The model sees only its context window, so most of a 400-pixel strip has no nearby real pixels to match against and invents freely — and the prompt still described a product, so it put one there. The fixes are steps of 128 to 256 pixels with 64 to 128 pixels of overlap into the existing image, and a prompt that describes what continues in the new area rather than what the picture is of.

2. Kiran protected an approved frame rather than regenerating. When would the other choice have been right?

When there is time. Outpainting is a rescue: it extends a scene and cannot extend a composition, and four careful steps can take longer than a fresh generation at the right ratio using the approved frame as a reference. With half a day rather than two hours, regenerating at 1344 × 768 and re-compositing would have given a composition designed for 16:9 instead of one stretched into it.

3. The extension was technically seamless and compositionally worse. What did that cost, and how was it recovered?

The original worked because the product held tension against the right edge; adding room removed it and left the product floating. It was recovered by cropping back into the enlarged canvas rather than using all of it — which was only possible because he had generated more room than the ratio strictly required. Judge an extended frame as a new picture, not only by whether the join shows.

MODULE 5 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. 407. If two go wrong, re-read the module before the exam.

1 Inpainting a hand that occupies eighty pixels keeps producing the same mush. Which setting changes the outcome?

A. Inpaint at full resolution, so the crop is regenerated at native size.
B. Increasing the mask blur so the new pixels blend into the old ones.
C. Adding "five fingers, anatomically correct" to the negative prompt.
D. Raising the denoise strength to 1.0 so that the region is completely replaced.

2 Why is grain always the last step in a repair sequence?

A. Grain raises the file size, so it should not be carried through the edits.
B. Upscalers remove grain, so adding it earlier wastes the pass.
C. Grain is the only step that cannot be undone non-destructively.
D. Applied before a repair, it leaves the repaired patch with none.

3 Why are eyes masked together rather than repaired one at a time?

A. One eye alone falls below the smallest mask the model will work with.
B. They must agree on gaze, size and catchlight, which separate passes cannot.
C. The face detector only triggers when both eyes are inside the mask.
D. A single mask is faster than two passes and costs fewer credits on a hosted service.

4 Extending a portrait sideways by outpainting produces a second person. Why?

A. The model mirrors the existing subject to fill a symmetrical canvas on both sides.

B. The overlap was too large, so the subject was regenerated twice.

C. The prompt still describes the person, and there is empty canvas.

D. The denoise was too low for the model to invent new background.

5 You are removing a cup from a table by inpainting. Which two adjustments make it work?

A. Mask tightly, use a denoise of 0.3, and let the surroundings bleed inward.

B. Mask the whole table, and prompt for the room to be regenerated.

C. Mask tightly around the cup, and add "no cup" to the prompt.

D. Mask its shadow and surroundings too, and prompt for the empty table.

6 Why is a product label composited in after a tiled generative upscale rather than put through it?

A. Tiles add plausible detail, so letterforms come back as different words.

B. The overlap region doubles the label's contrast at the seam.

C. Tiling downsamples a label to the tile's shortest edge.

D. A tiled pass strips the alpha channel that a cut-out label needs to sit on top.

6

MODULE 6

The same thing, twelve times

Deliver a set of images that hold one character, one product and one visual system — build a character sheet, choose between an image adapter and a trained LoRA on a stated cost-and-quality basis, prepare a training set that avoids the four common dataset faults, and hold light, palette, lens and finish constant across a campaign.

52 The same face twice is the hard part 272
53 Build the reference before you build the images 275
54 Training a character LoRA on hardware you can borrow 279
55 Four dataset faults, and the LoRA each one ruins 283
56 The decision, with the numbers 286
57 A product is harder than a face 290
58 Five constants that do more than the face 294
59 A file scheme that survives the revision round 299
60 A worked run, with the budget and the failure 304
Case study: Twelve frames, one weaver, and a photograph nobody had asked permission for 309
Module 6 test 314

Lesson 52 · 9 min

The same face twice is the hard part

THE ONE THING TO KEEP

One-photo identity adapters get you a family resemblance; a specific real person needs a LoRA trained on varied photographs, or their actual photographed face composited in an editor.

The requirement that separates demos from jobs

One striking image is easy now. Ten images of *the same person* in different scenes, or forty shots of *the* product with *the* label, is the requirement almost every real commission has, and it is genuinely hard.

There are four approaches. They are not alternatives to each other so much as rungs, and it is worth knowing exactly how far up each one gets you.

Rung one: seeds. Almost useless for this

A seed is the starting noise. Same seed, same model, same prompt, same settings gives you the same image, bit for bit. Change one word and the face changes.

So seeds reproduce an image, not an identity. They matter enormously for controlled experiments — change one clause, keep the seed, see what that clause did — and they matter for going back to a frame you liked. They do not give you the same person in a new pose. People waste days on this.

Rung two: identity adapters. Family resemblance

Feed one photograph of a face; get back a generated face that resembles it. This is IP-Adapter FaceID, InstantID and PuLID in the open world, and "character reference" or a trained identity object like Higgsfield's Soul ID on hosted platforms.

Honest ceiling: these carry the broad geometry of a face — the proportions, the general structure — and they are **family-resemblance accurate, not identity accurate**. The person's sister comes out. Skin texture drifts, apparent age drifts, and unusual angles break it. Expression is often flattened toward neutral.

That is fine for a stylised character, a game avatar, an illustrated persona. It is not fine when the client is looking at a picture of themselves, because humans have dedicated neural hardware for faces and will spot the wrongness without being able to name it.

The platform versions that ask for a set of photographs rather than one are doing something closer to rung three, and get closer to the person.

Rung three: a LoRA. The real answer

Train a small adapter on images of your subject. Twenty to thirty images, an hour or so of training, and the concept becomes something you can name in a prompt and place anywhere.

What actually determines whether it works:

- **Variety beats quantity.** Twenty photographs across different lighting, angles, distances and clothing beat two hundred taken in one session. If every training image was shot against the same blue wall, the LoRA has learned the blue wall as part of the person, and it will follow them into every scene.
- **Crop consistently and caption honestly.** Caption what varies, not what stays constant.
- **Stop early.** Overtraining bakes in the training set. The face becomes rigid and one of the photographs starts reappearing wholesale.
- **Free to do.** Kohya_ss, OneTrainer and ai-toolkit are free, and Kaggle's or Colab's free GPU hours are enough for a character LoRA. Fine-tuning covers the general method.

Rung four: do it in an editor

Generate the scene. Composite the real photographed head, or the real product, onto it. Match colour, match grain, match the light direction.

This is unglamorous and it is the most reliable thing in this lesson. Half of what looks like extraordinary AI work in commercial studios is a generated background with a photographed subject on top. Image editing is the skill that makes it invisible.

Products are harder than faces

A face can be five per cent wrong and read as the person. A logo cannot be five per cent wrong — a wrong letterform is simply a counterfeit. Bottle proportions, the exact green of a brand, the position of a seam: all of these must be exact and none of these are things a model holds precisely.

The professional answer: photograph the product, generate the environment, composite. Do not ask a model to draw a product that exists. Ask it to build a room to put the photograph in.

The consent line

A LoRA of a person is a tool for producing unlimited images of that person doing things they did not do. Training one on someone who has not agreed is not a technical decision, and in several countries it is now a legal one. Get permission in writing, and say what the images will be used for. Making things with AI sets out the wider consent and likeness position.

Today: if you have a recurring character, count your reference photographs. If they were all taken in one place on one afternoon, you do not yet have a training set — you have one photograph repeated.

BEFORE YOU MOVE ON

A client wants their founder to appear in six generated scenes. The designer uses a single-photo face reference feature and the client says the results look like the founder's cousin. What is the correct diagnosis?

A. The reference photograph is too low-resolution and a studio-quality headshot would fix it.

B. The seed should be locked across all six generations so the face stays constant.

C. Single-photo adapters transfer the broad geometry of a face and are family-resemblance accurate rather than identity accurate; a specific real person needs a LoRA trained on 20-30 varied photographs, or her real photographed head composited in.

D. The prompt needs to describe her features in detail alongside the reference image.

The answer and why: p. 414

Lesson 53 · 8 min

Build the reference before you build the images

THE ONE THING TO KEEP

A character sheet turns an identity from something you hope recurs into an asset you condition on, and building it first is what makes every later frame a matter of reuse rather than of luck.

Working backwards from the problem

You need one person across eight images. The naive approach generates image one, likes the face, and then tries to describe that face well enough to get it again. It does not work, and the reason is simple: a face is not

describable at the precision required. "Woman, 30s, dark hair, round face, brown eyes" describes several million people, and the model will show you a different one each time.

The method that works is to stop trying to describe and start building a **reference sheet**: a small set of images of the same person from several angles, in neutral conditions, that you use as input for everything afterwards.

Animation and games have worked this way for a century. The character sheet comes before the scenes.

What goes on the sheet

Five to eight images of one person:

- Front, neutral expression, even light
- Three-quarter left, three-quarter right
- Profile
- One at a distance, full body, showing proportions and posture
- Two with expression — a smile, a look away
- One in the project's actual lighting, which becomes your style anchor

Neutral is the point. Flat even light, plain background, no strong grade. You want the identity separated from everything else, because everything else will change in the final frames.

Getting the first face

You need one image of somebody who does not exist. Three routes:

Generate and select. Run a contact sheet of thirty portraits with a description, pick the one you want. Free, fast, and the face is the model's average of your words, which sometimes means it is an unremarkable face.

Generate and refine. Pick a face, then use seed variation at strength 0.1–0.2 to explore its neighbourhood until it is specific rather than generic. Worth the extra ten minutes, because a distinctive face holds better across images than an average one — there is more for the mechanism to lock onto.

Photograph a consenting person and stylise. Legitimate with permission, and the identity block later in this course sets out why permission is not optional.

Building the other angles

Once you have the front view, you need the rest, and they must be the same person.

1. Take the front view as an IP-Adapter face reference at weight 0.7–0.85.
2. Add a pose control with the target angle.
3. Generate. Check identity. Adjust adapter weight.
4. Repeat for each angle.

Expect drift. The profile is the hardest — there is least information about a profile in a front view — and it often takes eight attempts. Accept a close match rather than a perfect one; the sheet's job is to give a *combined* signal, and small variation across it is tolerable and sometimes helpful.

Using the sheet

Two ways, and you will use both.

As adapter input. Feed two or three sheet images as face references for each new generation. Fast, needs no training, and gives a family resemblance that holds well enough for many jobs.

As LoRA training data. The sheet is a training set. Fifteen to thirty images trains a character LoRA that holds identity far more tightly. The next two lessons cover it.

The efficient path is to build the sheet, use adapters for the first few frames to check the character works in context, and train a LoRA only if the job needs more than an adapter gives.

The limit, stated honestly

A character sheet gets you a person who is *recognisably the same* across images. It does not get you a person who is *identically the same*. Bone structure holds; small features drift. The nose is slightly different. The distance between the eyes moves by a pixel or two. In a set viewed one image at a time, nobody notices. In a set viewed side by side at large size, some people do.

The gap narrows with a well-trained LoRA and narrows further with careful repair, and it does not close. If your brief genuinely requires an identical face — a recurring character in a long campaign, a brand mascot that will be scrutinised — budget for face compositing on top of everything else: generate the body and scene, and place a consistent face in from a fixed set, matched for angle and light.

That is more work and it is the only method that is actually reliable. Knowing this at the brief stage, rather than at frame nine, is the difference between a plan and a crisis.

BEFORE YOU MOVE ON

Why build a character sheet before generating any of the final frames?

A. Because an identity cannot be described in words precisely enough, so it has to become an image you condition on.

B. Because generating the sheet first establishes a consistent seed that can then be reused across every frame in the set in order to hold the face stable.

C. Because a sheet built from the final frames would inherit their lighting and grade, which contaminates the identity signal used for subsequent conditioning.

D. Because training a LoRA requires images at several angles, and those angles cannot be produced once the character has already appeared in finished work.

The answer and why: p. 414

Lesson 54 · 9 min

Training a character LoRA on hardware you can borrow

THE ONE THING TO KEEP

A usable character LoRA trains from 15 to 30 images in under an hour on free cloud hardware, and the settings that matter are rank, learning rate and total steps, with overfitting visible as the training images reappearing in the output.

What you are actually doing

A LoRA adds small trainable matrices alongside the model's existing weights and trains only those. The base model is untouched. That is why the result is 20–150 MB rather than several gigabytes, and why it trains in minutes rather than weeks.

For a character, you are teaching the model to associate a rare token — `ohwx`, `sks`, or any string it does not already have a meaning for — with a specific face.

The recipe

Reasonable starting settings for an SDXL character LoRA:

```
images:            20  (15 minimum, 30 is plenty)
resolution:        1024
repeats:           10
epochs:            10
total steps:       20 x 10 x 10 / batch 2  = 1000
network rank:      16      (8 for a simple style, 32 for a
complex identity)
network alpha:     8       (half the rank is a common
convention)
learning rate:     1e-4    (unet), 5e-5 (text encoder)
scheduler:         cosine with 10% warmup
optimiser:         AdamW8bit
save every epoch:  yes
```

Save every epoch. This is not optional advice — it is how you find the right amount of training. You will end up with ten files and the best one is usually not the last.

Running it free

Kaggle Notebooks give about 30 GPU-hours a week, enough for many training runs. **Google Colab** free tier works for SD 1.5 comfortably and SDXL with care. Both have ready-made notebooks for the common trainers — Kohya's scripts and OneTrainer are the usual choices, both free and open source.

Expect, for 1000 steps at 1024 resolution: roughly 25–40 minutes on a T4, 12–20 on a better card. Session limits are the real constraint, not capability. Upload the dataset to the session, train, and download the resulting files before the session ends, because it will end.

Locally, 12 GB of VRAM trains SDXL LoRAs comfortably; 8 GB works with gradient checkpointing and a batch size of 1, slowly.

Finding the right epoch

Generate the same prompt and seed with each saved epoch and lay them out as a grid.

- **Early epochs (1–3)**: identity weak, prompt followed well.
- **Middle (4–7)**: identity present, prompt still working. **This is where the good one is.**
- **Late (8–10)**: identity strong, prompt increasingly ignored, backgrounds from your training images appearing.

That last symptom is the definition of overfitting, and it is unmistakable once you know it: you ask for your character on a beach and get your character in front of the grey wall that was in twelve of your training photographs. The model has learned the wall as part of the concept.

Pick the middle epoch. Almost everyone's first LoRA is overtrained, because more training feels like more learning.

Captions

Each training image gets a caption. The rule is counter-intuitive and worth stating clearly: **caption what should vary, not what should stay.**

```
ohwx woman, three-quarter view, grey background, neutral
expression
ohwx woman, smiling, outdoors, blue jacket
```

The token `ohwx` is constant, so the model attaches the invariant part — the face — to it. Everything you *do* caption becomes separable and controllable afterwards. If you caption the jacket, you can change the jacket later. If you do not, the jacket becomes part of the character.

This is the single most useful thing to understand about captioning, and it explains a whole class of "my LoRA always puts her in the same shirt" complaints.

Style LoRAs differ

For a style rather than a character: 30–100 images, lower rank (8–16), fewer steps, and captions that describe *content* thoroughly while never naming the style. The logic is the same — the style is the invariant, so leave it uncaptioned

and it attaches to the trigger token.

The honest assessment

A character LoRA takes about two hours end to end the first time: 30 minutes assembling and captioning, 30 training, 30 testing epochs, 30 tuning strength in real prompts. After a few, it is under an hour.

It gives you an identity that holds across dozens of images, works with any prompt, composes with controls, and is a file you own. For a campaign, a recurring character, or anything you will return to, it repays the hour many times.

It is the wrong answer for one image, and it is the wrong answer when you have not first checked that an adapter is good enough. And it has a specific failure worth knowing: a LoRA trained on 20 images of one face in similar conditions will reproduce those conditions, so the identity arrives carrying the lighting it was trained under. That is what the next lesson is about, and it is the difference between a LoRA that works and one that does not.

BEFORE YOU MOVE ON

A character LoRA reproduces the grey wall from its training photographs no matter what environment the prompt asks for. What has happened?

A. The network rank is too high, giving the adapter more capacity than a single face requires and allowing it to memorise incidental detail.

B. The learning rate was set too high, so each update moved the weights far enough to capture the background alongside the subject.

C. Overfitting: the wall was constant across the training set and never captioned, so it became part of the concept attached to the trigger token.

D. The trigger token chosen was not sufficiently rare, so it carried a prior association with plain studio backgrounds from the base model's own training.

The answer and why: p. 415

Lesson 55 · 8 min

Four dataset faults, and the LoRA each one ruins

THE ONE THING TO KEEP

A LoRA learns whatever is constant across its training images, so variety in everything except the subject is the property that decides whether it works, and no training setting compensates for a set that lacks it.

The rule that explains every failure

A LoRA learns what does not change.

That is the whole thing. If the face is the only constant, the face is what it learns. If the face *and* the lighting *and* the background *and* the camera angle are all constant, it learns all four as one inseparable concept, and you get a LoRA that can only produce your character in that light, in that place, from that angle.

Every dataset fault is a violation of this rule.

Fault one: no variety in lighting

Twenty images shot in the same session under the same light. The LoRA now believes soft frontal window light is part of the person′s identity. Prompt for dramatic side lighting and you get a fight between the LoRA and the prompt, which the LoRA usually wins, producing a flat face awkwardly embedded in a dramatic scene.

Fix: vary it deliberately. Some soft, some hard, some outdoors, some indoors, some backlit. If you are generating the training images from a character sheet, generate them under different lighting on purpose.

Fault two: no variety in angle and distance

All front-facing at the same crop. The LoRA produces a good front view and falls apart in profile, because it has never seen one.

Fix: roughly half front and three-quarter, a quarter profile or near-profile, a quarter at greater distance including full body. The distant ones matter more than people expect — without them, the LoRA has no idea what the character's proportions are and will produce a correct face on an arbitrary body.

Fault three: near-duplicate images

Five frames from the same burst, or the same image cropped five ways. These are nearly identical, so they count five times toward whatever they have in common, and the set is effectively much smaller than its file count.

Fix: one image per moment. Fifteen genuinely different images beat forty near-duplicates comfortably. If you are unsure, lay the set out as a contact sheet and look for anything that reads as the same picture twice.

Fault four: other people, or other versions of the subject

A second face in the background. A reflection. A photograph of the person at a very different age or with a very different haircut, mixed into a set that is otherwise consistent.

Fix: crop out other faces or discard the image. Pick one era and stay in it — a LoRA trained across a fifteen-year span learns an average that resembles neither.

The quality floor

Resolution: at least your training resolution, so 1024 for SDXL. Upscaling a small image to meet it teaches the model upscaling artefacts as part of the identity.

Sharpness: avoid heavily blurred or motion-blurred images. Some softness is fine and even useful, but a set full of soft images produces a soft LoRA.

Compression: avoid heavily compressed JPEGs. The blocking gets learned. This matters more than it sounds, and it is why a LoRA trained from images saved off a messaging app often looks subtly bad in a way nobody can identify.

A checklist before you train

Lay the set out as a grid and ask:

1. Is the subject the only thing constant across all of these?
2. Are there at least three distinct lighting situations?
3. Are there at least three distinct angles, including one distant?
4. Would any two of these be described as the same picture?
5. Is there another face anywhere in frame?
6. Is every image at or above training resolution, and reasonably sharp?
7. Do the captions name everything that varies and nothing that should not?

Ten minutes on this checklist saves the two hours of training and testing a bad LoRA and then working out why it is bad.

The honest constraint

You cannot always get variety. Sometimes you have eight photographs of a real person, all from one afternoon, and that is all that exists. Train on them anyway — a limited LoRA is better than none — but know what you have: a LoRA that works in that lighting and fails elsewhere, and a job where you should plan to match the scene lighting to the training lighting rather than the other way round.

And a note that belongs here rather than in the legal block, because it is a working decision rather than a legal one. If the training images are of a real person, you need their agreement, and "I found them online" is not agreement. The images also carry the rights of whoever took them. A dataset assembled from a search engine is a problem you are building into an asset you intend to reuse, and it is much cheaper to resolve before training than after a campaign has shipped.

BEFORE YOU MOVE ON

Why does a training set shot entirely in one lighting setup produce a LoRA that resists prompted lighting changes?

A. Because the adapter's weights are optimised for the tonal range present in the training images, so other ranges fall outside what it can reproduce.

B. Because lighting information is encoded in the same layers as identity, so the two cannot be separated by any captioning strategy.

C. Because a LoRA learns whatever is constant, so with lighting held fixed it becomes part of the concept rather than a separable attribute.

D. Because the captions described the subject but not the lighting, and any attribute left uncaptioned is excluded from the learned concept entirely.

The answer and why: p. 415

Lesson 56 · 7 min

The decision, with the numbers

THE ONE THING TO KEEP

An adapter gives roughly seventy per cent of an identity in under a minute and a trained LoRA gives most of the rest in about an hour, so the decision turns on how many images the identity has to survive and how closely it will be examined.

Stop treating it as a preference

People adopt one method and defend it. It is a decision with two inputs: **how many images**, and **how closely will it be looked at**.

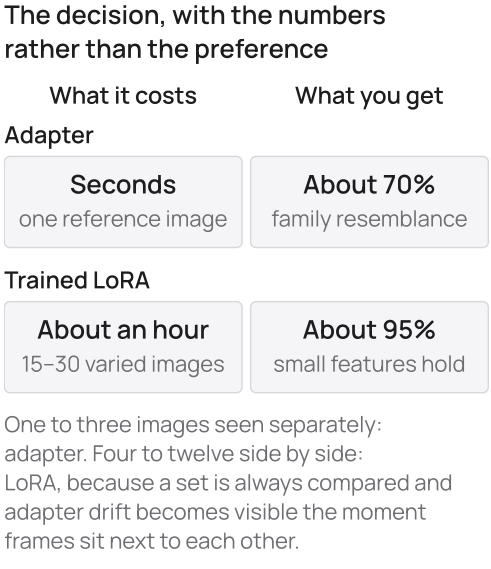
The comparison, honestly

Image-prompt adapter (IP-Adapter and relatives)

- Setup: seconds. Load a reference, set a weight.
- Cost: nothing beyond generation.
- Result: a family resemblance. Bone structure and general appearance carry; small features drift between images.
- Flexibility: works with any prompt, though high weights start to compete with it.
- Requires: one image.

Trained LoRA

Figure 6.1



One to three images seen separately: adapter. Four to twelve side by side: LoRA, because a set is always compared and adapter drift becomes visible the moment frames sit next to each other.

- Setup: about an hour end to end, once you have done it before.
- Cost: free on Kaggle or Colab; a few pennies of electricity locally.
- Result: identity holds tightly, including small features, across dozens of images.
- Flexibility: composes cleanly with prompts and controls at sensible strengths.
- Requires: 15–30 varied images, which for a fictional character means building the sheet first.

The decision rule

One to three images, seen separately — adapter. The hour of training buys nothing a viewer will register.

Four to twelve images, a campaign, side by side — LoRA. Adapter drift becomes visible when frames are compared, and a set is always compared.

A recurring character across months — LoRA, without question. It is an asset you keep, and the alternative is re-deriving the character every time.

Close scrutiny at large size — LoRA, plus face compositing for the frames that matter most.

You do not yet know if the character works — adapter first. Test the concept cheaply, then train once it is approved. Training a LoRA for a character the client then rejects is an hour spent on nothing.

Using both

The strongest arrangement for a demanding set is both together: the LoRA at 0.7–0.8 carrying the identity, and a face adapter at 0.3–0.4 from the single best reference nudging it toward one specific rendering.

The caution is the stacking budget from the previous block. A LoRA and an adapter are both constraints, and adding a pose control and a depth control on top leaves nothing free. If four seeds look identical, drop something.

What neither solves

Both fix *who the person is*. Neither fixes *how they are lit, framed or graded*, and those are what make a set look like a set. A perfectly consistent face across twelve differently-lit, differently-framed images still reads as twelve unrelated pictures.

That is the next lesson, and it is the part people skip. Identity consistency is the technically interesting problem, so it absorbs all the attention, while the visual system — one light, one lens, one palette, one grain — is what a viewer actually perceives as coherence and costs almost nothing to enforce.

Put differently: a set with a slightly drifting face and a rigorously consistent look reads as consistent. A set with a perfect face and twelve different lighting setups does not. If you have limited time, spend it on the system.

Testing which one you need, cheaply

You do not have to guess. The test takes fifteen minutes and settles the question with evidence rather than opinion.

Generate four frames with an adapter at 0.75 — the four most different situations the brief contains, so different angles, different distances, different lighting. Lay them out side by side at the size they will be delivered.

Then ask one question: **would a stranger say this is the same person?** Not "can I see the differences", because you can, having stared at them. A stranger, at delivery size, in the order the audience will see them.

If yes, ship the adapter and keep the hour. If no, train. If you are unsure, show it to somebody who has not been in the project — the answer is usually immediate and it is usually right.

Strength values worth starting from

For an adapter: 0.6–0.8 for a face. Below 0.5 the identity barely registers; above 0.9 it starts overriding the prompt and the pose.

For a LoRA: 0.6–0.85, depending on how heavily trained it is. A LoRA that needs 1.0 to show up is undertrained; one that dominates at 0.5 is overtrained, and the middle epoch you did not pick is probably the right one.

Write the working value next to the LoRA's filename, because you will not remember it in a month and rediscovering it costs a dozen renders.

BEFORE YOU MOVE ON

When is an image-prompt adapter the right choice over a trained LoRA?

A. When the base model already renders the desired character type well, so only a light adjustment toward the specific reference is required.

B. When the character appears in one or two images seen separately, where drift between frames will never be observed.

C. When the character must appear alongside controls such as pose and depth, since a LoRA cannot be combined with structural conditioning in the same generation.

D. When there are fewer than fifteen reference images available, which is the minimum a LoRA needs in order to train at all.

The answer and why: p. 416

Lesson 57 · 8 min

A product is harder than a face

THE ONE THING TO KEEP

A face is forgiven small variation and a product is not, because a viewer compares the rendered product against the real one they can buy, so compositing a photograph is almost always the correct answer.

Why the same techniques fail here

Everything in this block works better on people than on objects, and the reason is about the viewer rather than the model.

A face has natural variation. It changes with expression, angle and light, so a viewer's tolerance for small differences is wide — they are used to the same person looking different in different photographs.

A product does not vary. It is a manufactured object with fixed proportions, a fixed label, a fixed logo. Any difference between two images is a defect, and the viewer can check it against the actual thing on a shelf.

So the tolerance that makes character LoRAs acceptable does not exist here.

What goes wrong specifically

The label. Text, which the repair block established cannot be generated reliably. A near-correct label is worse than none: it reads as a counterfeit.

Proportions. A bottle's neck-to-body ratio drifts a few per cent between images. Individually imperceptible, obvious side by side, and immediately wrong to anybody who knows the product.

The logo. A trademark, which must be exact. An approximation is a legal and commercial problem, not an aesthetic one.

Material behaviour. How glass refracts, how a metallic finish catches light, how a matte surface falls off. Generated versions are plausible and not specific.

Colour. As the brief block covered, a specific colour is not something a diffusion model can hit.

The answer, and it is not a compromise

Photograph the product. Generate the environment. Composite.

The full method is in the previous block. Here is why it is specifically the right answer for consistency: a photograph of a product is *identical every time you use it*. Twelve images using the same cut-out are twelve images with exactly the same product, at zero risk. No LoRA, no drift, no label problem.

For a set, shoot the product once properly at several angles, build a small library of clean cut-outs, and reuse them across every frame. That library is an asset that serves this campaign and the next three.

When you must generate the product

A product that does not exist yet — a concept, a pitch, a packaging proposal. Then:

1. **Model it, if you can.** A simple shape in Blender takes an hour and gives you exact, identical geometry from any angle, with the label applied as a texture. This is the right answer for anything with a defined form, and it is free.
2. **Train a LoRA on renders.** If you have a 3D model, render 30 views and train. Now the product is in the model's vocabulary and can appear in generated scenes with reasonable consistency.
3. **Generate once, then reuse.** Create the product image once, cut it out, and composite it everywhere — treating your own generated product exactly as you would a photographed one.

That third option is the one people miss, and it is usually the best. The moment you have one acceptable version of the object, stop generating it. Turn it into an asset.

The label, always

Whatever route you take, the label is set as type and applied afterwards. Generate or model the blank container, then place the artwork on it with a perspective transform, matched for lighting and curvature.

For a curved surface, the free path is a displacement map in GIMP: build a greyscale map of the curvature, apply it to the flat label layer, then multiply it over the product so the underlying shading shows through. Fifteen minutes, and it holds up at print size.

The exception worth knowing

Generated products are fine where the product is **generic and not the subject of a claim** — a coffee cup on a desk in a lifestyle image, an unbranded notebook, a plant pot. These are set dressing, and nobody will compare them against anything.

The line is whether the object is what the image is *about*. Set dressing can be generated. The hero product cannot, and pretending otherwise produces a specific, avoidable embarrassment: a client seeing their own product rendered with a label that nearly says the right thing.

Building the cut-out library properly

Since the photograph is the asset, shoot it once and shoot it well. An afternoon spent here serves every campaign afterwards.

For each product:

- **Four to six angles**: front, three-quarter left and right, a slight top-down, and one at the height the product usually sits on a table.
- **Two lighting setups**: one soft, one with a harder directional key, so you can match either kind of generated scene without reshooting.
- **A clean background**: mid-grey is better than white or black, because it contaminates the cut-out edge far less than either extreme.
- **The product turned so the label is legible** in at least half the frames.

Cut them out, save them as PNGs with transparency, and store them in a `product/` folder alongside a short text file saying what the lighting was in each. That note is what lets you pick the right cut-out in eight seconds three months later rather than opening all twenty.

The reflection problem

One thing worth planning for. A product placed on a glossy generated surface needs a reflection, and the reflection has to come from your cut-out — a flipped, faded, slightly blurred copy on a layer beneath it, masked so it fades with distance.

If the surface is strongly reflective, the product also needs to reflect *the scene*, which the photograph obviously does not contain. For a matte product this is negligible. For chrome or glass it is the whole job, and it is the case where a 3D render genuinely beats a photograph, because a renderer can reflect an environment map you supply.

BEFORE YOU MOVE ON

Why is a product harder to keep consistent across a set than a human character?

A. Because manufactured objects have hard edges and regular geometry, which diffusion models reproduce less accurately than the organic forms of a face.

B. Because a product is fixed and checkable against the real item, so variation reads as a defect rather than as natural difference.

C. Because a product's label contains text, and text is the failure a generative model cannot reliably avoid in any part of an image.

D. Because character consistency techniques such as LoRAs are trained predominantly on human faces and transfer poorly to inanimate subjects.

The answer and why: p. 416

Lesson 58 · 8 min

Five constants that do more than the face

THE ONE THING TO KEEP

Coherence in a set is produced by holding light, lens, palette, crop and finish constant, and these are cheap to enforce and far more visible to a viewer than small drift in a character's features.

What a viewer actually perceives

Show somebody twelve images and ask whether they belong together. They are not examining the nose. They are responding to whether the images look like they were made by one person, in one session, with one intent.

That perception is built from five things, all of which you control completely.

The five constants

1. Light. One direction, one quality, one colour temperature across every frame. Hard sun from the left in all twelve, or soft window light in all twelve. Mixing them is the single strongest signal that a set is not a set.

2. Lens. One focal length and one camera height. A 35mm waist-level look throughout. Mixing a 24mm wide with an 85mm portrait reads as two different shoots regardless of anything else.

3. Palette. Three or four colours that recur, and one that does not appear anywhere. A restriction is more visible than an inclusion: if nothing in the set is pure saturated red, the set acquires a discipline viewers feel without identifying.

4. Crop convention. How much room around the subject, where the horizon sits, whether subjects are centred or offset. Consistency here is what makes a grid of images feel designed.

5. Finish. Grain, contrast curve, colour grade, vignette. Applied identically to every frame as the last step. This is the cheapest of the five and possibly the most effective.

Figure 6.2

What a viewer perceives as a set, and what absorbs the effort

What actually holds a set together

- One light direction, quality and temperature
- One focal length and one camera height
- Three or four recurring colours, and one absent
- One crop convention, held across every frame
- One grade and grain, applied last and identically

What people spend the time on instead

- Identity, because it is the interesting problem
- A face that holds to the pixel
- Twelve individually beautiful frames
- Judging each image on its own
- Regrading by eye, frame by frame, at the end

A set with a slightly drifting face and a rigorous look reads as consistent. A set with a perfect face and twelve lighting setups does not. If time is short, spend it on the left.

Enforcing them

Write them down as text, in the project folder, and reuse the text:

```
# system — spring campaign
LIGHT    hard sun from camera left, late afternoon, long shadows
LENS     35mm, waist level, slight wide
PALETTE  terracotta, sage, bone white, deep brown.  No blue.
CROP     subject in left third, horizon at 2/3, generous
headroom
FINISH   colour negative grain, lifted blacks, warm mids, -10
saturation
```

Slots five, six and seven of every prompt come straight from this file. The finish is applied as one saved adjustment set — a GIMP script, an exported curve, or a LUT — over every flattened frame.

The anchor image

Pick or build one frame that embodies the system completely, finish it fully, and get it approved. It then does three jobs:

- **Editorially**, it is what the client signed off on.
- **Technically**, it is a style reference for an adapter and a colour-match target for grading.
- **Practically**, it is the comparison for every later frame. Does frame nine belong next to the anchor? If not, frame nine changes.

The final pass that people skip

When all twelve exist, put them on one screen at the same size, in the order they will be seen. Do not skip this because they each looked right on their own.

You will find: one frame warmer than the rest; one with the light from the wrong side; one cropped tighter; one with more contrast. Every set has them, and they are invisible one at a time and obvious together.

Fix them globally, not individually. A warm frame is one curves adjustment. A frame with different contrast is one curve. These are two-minute corrections at this stage and would have been twenty-minute regenerations earlier.

The deliberate exception

A set does not have to be uniform, and a set where every frame is identical in treatment can be monotonous. What it needs is a *system*, which can include variation that is itself systematic: three close-ups among nine wides, one dark frame among eleven light ones, a rhythm rather than a drone.

The distinction is between variation you chose and variation that happened. A dark frame placed deliberately at the midpoint of a sequence reads as intent. A dark frame because that generation came out dark reads as a mistake.

So when the final pass finds an outlier, the question is not only "is this different" but "is this different on purpose". Occasionally the answer is yes and you keep it. Usually the answer is no, and one curves adjustment fixes it.

Making the finish reusable

The fifth constant is the one you can automate, and automating it is worth twenty minutes once.

Build the grade on the anchor image, then save it in a form you can reapply:

- **GIMP**: export the curve from the Curves dialog as a settings file, and save the grain layer as a reusable XCF you paste in.
- **Krita**: save the adjustment stack as a layer style, or keep a template document.
- **Any tool**: export a LUT from the anchor and apply it everywhere. Free LUT generators exist, and a LUT is the most portable form — it will still apply in five years in software that does not exist yet.

Then the finish step on every frame is: flatten, apply LUT, add grain layer, done. Two minutes rather than fifteen, and identical by construction rather than by care.

What to do when the client changes a constant

They will, usually after three frames are finished. "Can it be cooler?"

Because the constants live in one file and the finish lives in one preset, this is a small change: edit the system file, rebuild the preset, reapply to the finished frames. Twenty minutes for the whole set.

Without that structure, it is a per-frame regrade done by eye, which takes an afternoon and produces twelve frames that are each individually cooler and no longer identical to each other. The system file is not administration; it is what

makes a late change survivable.

BEFORE YOU MOVE ON

Why does holding five visual constants matter more to a set's coherence than perfect character consistency?

A. Because character drift is only visible at large sizes, whereas lighting and palette inconsistencies remain visible even in thumbnails.

B. Because viewers judge whether images were made together from light, lens, palette, crop and finish rather than from small facial differences.

C. Because the five constants can be applied after generation, which makes them the only aspects of a set that can still be corrected once the frames exist.

D. Because a set is usually viewed as a sequence rather than side by side, and sequential viewing hides facial variation while exposing tonal shifts.

The answer and why: p. 417

Lesson 59 · 7 min

A file scheme that survives the revision round

THE ONE THING TO KEEP

A naming convention that encodes project, frame, version and status lets you find the approved file in seconds six months later, and its absence is what turns a small revision into an afternoon.

The situation it prevents

Three weeks after delivery, a client asks for frame seven with the warmer grade "like you showed us the first time". Your folder contains `hero_final.png`, `hero_final_v2.png`, `hero_FINAL_new.png`, `hero7.png`,

`hero7 copy.png` and `Untitled-3.xcf`. Finding the right one takes forty minutes and you are not certain when you stop.

This is entirely avoidable and the fix takes no discipline once it is a habit.

A scheme that works

```
<project>-<frame>-<stage>-v<nn>[-<status>].<ext>

spring26-07-gen-v01.png          raw generation, metadata intact
spring26-07-gen-v02.png          second base render
spring26-07-repair-v01.xcf       layered repair file
spring26-07-final-v03-APPROVED.jpg
spring26-07-final-v04.jpg
```

Four rules make it work:

1. **Never overwrite.** New version, new number. Disk is cheaper than your time.
2. **Never use the word "final" without a version number.** It is the most broken word in creative work.
3. **Mark approval in the filename.** `APPROVED` is searchable across the whole disk in one command.
4. **Date the folder, not every file.** `2026-03-spring-campaign/` puts the date where it is useful.

The folder structure

```
spring26/
  brief.md                    the spec, with its change log
  ENVIRONMENT.txt             interface commit, package versions
  models.txt                  every model with its hash
  system.md                   the five constants
  refs/                       the labelled reference pack
  product/                    photographed cut-outs
  gen/                        raw generations, never edited
  work/                       layered editor files
  deliver/                    only files actually sent
```

Two rules make it survive: `gen/` is read-only in practice, and `deliver/` contains only what was really delivered. If those hold, the folder answers any question anybody asks later.

The seed log

Alongside it, one plain text file per project:

```
frame 07  courtyard
  base      spring26-07-gen-v02   seed 418 / dpmpp_2m karras /
26 / cfg 5.5
            model realvis-xl-v5 (a1b2c3d4)
            lora  char-meera-e06 @ 0.75
  repair    hands x2, doorway, bangle
  grade     warm curve + grain preset "campaign-a"
  sent      v03 on 18 Mar, APPROVED 19 Mar
  note      client preferred warmer grade over v02
```

Ten lines per frame. Written as you go, it costs about a minute; reconstructed later, it costs an afternoon and is usually impossible.

That last line is the one that pays off most. Three weeks later, "client preferred warmer grade" is exactly the fact you need and exactly the fact nobody remembers.

What to keep and what to delete

Keep: every approved file, every layered work file, the raw generation behind every approved image, the brief, the system, the logs, the environment and model files, and any community model a delivered image depended on.

Delete without hesitation: the other 400 exploration renders, superseded versions more than two rounds old, intermediate upscales. These are recoverable from the seed log if you ever need them, which you will not.

A hero image project typically leaves 200–400 MB worth keeping, once the exploration is cleared. That fits any backup.

The honest limit

None of this survives if it is not automatic, and a scheme that requires thought at every save will be abandoned by Thursday. Make it mechanical: set the interface's output template once so generations are named correctly without you doing anything, and keep the folder structure as a template you copy at the start of each job.

The measure of whether it works is a single question: **can you find the approved file for frame seven in under thirty seconds, six months from now?** If yes, the scheme is adequate however untidy it looks. If no, no amount of elegance in the naming convention is helping you.

Setting the output template once

Most interfaces let you control how generated files are named, and configuring it is a five-minute job that removes the problem at source.

In A1111 and Forge the setting is under output filename patterns, and something like this is useful:

```
[datetime<%Y%m%d>]-[seed]-[model_name]
```

In ComfyUI the Save Image node takes a filename prefix, so set it per project — `spring26-07` — and every render from that graph is already labelled with the frame it belongs to.

The point is that a filename produced automatically is always correct, and one you type at save time is correct until the afternoon you are tired.

Version numbers the client can use

Use the same version number in the filename and in the message you send. "Attached: v03" and a file called `spring26-07-final-v03.jpg` means their reply of "v03 is approved, but can we see v02's grade again" is unambiguous.

Clients will otherwise say "the second one you sent", which is a description of your outbox rather than of a file, and reconstructing what that was takes longer than it should.

Two further habits that cost nothing. Never reuse a version number, even for a file you sent and immediately replaced — go to v04. And never send a file with `copy` or a space-number suffix in its name, because whatever you send is what they will quote back at you.

BEFORE YOU MOVE ON

What is the practical test of whether a file naming and folder scheme is good enough?

A. Whether every file in the project can be traced back to the generation parameters that produced it through the recorded metadata.

B. Whether the approved file for a given frame can be located in under thirty seconds several months later.

C. Whether the structure is consistent enough that another person could take over the project without being given any explanation of it.

D. Whether every intermediate version has been retained, so that any earlier state of the work can be restored if a client changes their mind.

The answer and why: p. 417

Lesson 60 · 8 min

A worked run, with the budget and the failure

THE ONE THING TO KEEP

A twelve-image campaign is about thirty hours of work distributed unevenly — a third on establishing the system, most of the rest on repair and consistency, and very little on generation itself.

The brief

A regional food brand. Twelve images for a spring campaign: four hero frames for print, eight social assets. One recurring character, a cook. One product, a jar, which exists and can be photographed. Deadline three weeks.

This is a realistic job and the numbers below are realistic.

Days one and two: the spec and the system

Write the spec. Placements and pixel sizes for all twelve. The forbidden list. Two rounds of revisions. Disclosure agreed.

Build the reference pack with the client — three directions, they choose one.

Photograph the jar. Half a day: window light, white card, four angles, thirty frames each. Cut out the best four. This library serves the entire campaign and every future one.

Write the system file: light, lens, palette, crop, finish.

Elapsed: roughly 8 hours. Generated images: about 20.

Days three and four: the character and the anchor

Generate a contact sheet of 30 portraits. Pick one, refine with seed variation. Build a character sheet of seven angles using a face adapter. Two hours.

Decide: twelve images, seen side by side, close scrutiny in print. Train a LoRA. Assemble and caption 22 images from the sheet, train on Kaggle, test the epochs, pick epoch six. Two hours including waiting.

Build the anchor frame: contact sheet, keeper checklist, full repair, grade, grain. Send for approval.

Elapsed: about 10 hours. Generated: about 150.

Day five: approval, and a change

The client approves the anchor but wants the palette warmer and the character a little older.

The warmer palette is a curve change, applied to the finish preset. Ten minutes, and it now applies to everything not yet made.

The character being older is the expensive one. It means a new character sheet and a retrained LoRA — about three hours. This is why the anchor is approved before the other eleven exist: at frame eleven this would have cost three days.

Elapsed: 3 hours.

Days six to eleven: the other eleven frames

Per frame, roughly:

- Contact sheet of 16 at low steps, pick two. *(5 min)*
- Full-quality renders of both with the LoRA, system prompt, and controls where needed. *(5 min)*
- Keeper checklist, repair list. *(5 min)*
- Repair: hands, face, edges, details. *(30–50 min)*
- Composite the jar where it appears, with contact shadow and grain. *(20 min)*
- Grade and grain from the preset. *(5 min)*

About 80 minutes a frame, so around 15 hours for eleven. The hardest frame — two people, hands passing the jar — takes three hours on its own and is done second rather than last.

Elapsed: about 15 hours. Generated: about 400.

Day twelve: the consistency pass

All twelve on one screen. Three are off: one warmer, one with the light from the wrong side, one cropped tighter.

The warm one and the tight crop are quick. The one lit from the wrong side is not fixable and is regenerated — two hours. This is a normal outcome and is why the schedule has slack in it.

Elapsed: about 4 hours.

Delivery

Export every size. QA pass against the forbidden list on each of twelve files. Alt text. Delivery folder, disclosure note, archive of the layered files and the LoRA.

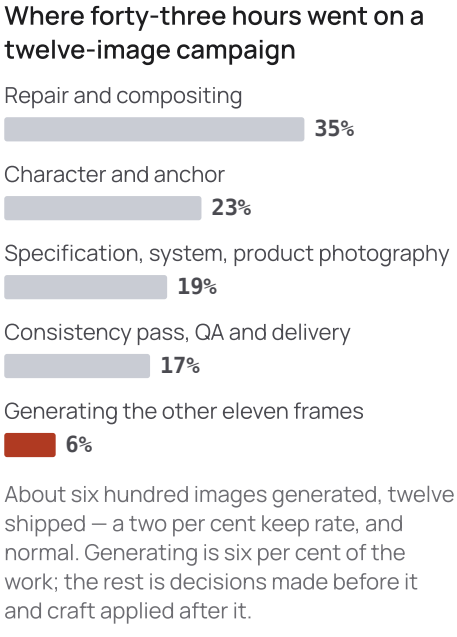
Elapsed: about 3 hours.

The totals

About 43 hours. Roughly 600 generated images, of which 12 shipped. That is a 2% keep rate, and it is normal — most of the 600 were low-step contact-sheet frames costing seconds each.

The distribution is the useful part:

Figure 6.3



- Specification, system, product photography: **19%**
- Character and anchor: **23%**
- Generating the other eleven: **6%**
- Repair and compositing: **35%**
- Consistency, QA, delivery: **17%**

Generation is six per cent of the work. Everything else is decisions made before it and craft applied after it, which is the argument this whole course has been making.

What actually goes wrong

Not the generating. In order of frequency: a change of direction after several frames are finished, an element that turns out to be ungeneratable and needs photographing, consistency problems found in the final pass, and the client wanting a size nobody mentioned at the start.

All four are addressed by the same two habits — approve the anchor before building the set, and do the hardest frame second. Neither is technical, and together they prevent more lost days than any setting in this course.

BEFORE YOU MOVE ON

In a realistic twelve-image campaign, roughly what share of the total time is spent generating the non-anchor frames?

A. Around a third, since each of the eleven frames requires its own exploration and several full-quality candidate renders before one is chosen.

B. Around a fifth, because generation and repair together dominate a production of this size once the system has been established.

C. Under ten per cent, because generation is fast once the specification, system, character and anchor are settled.

D. About half, since generation is the only stage that must be repeated for every frame in the set rather than done once.

The answer and why: p. 418

CASE STUDY · MODULE 6

Twelve frames, one weaver, and a photograph nobody had asked permission for

An illustrative case, built from documented patterns.

A handloom co-operative in Madurai commissioning a twelve-image festival campaign with one recurring character.

The campaign was agreed at twelve images: four for print, eight for social, all sharing one visual system and one recurring character — a weaver at a pit loom, seen across a working day. The co-operative had three weeks and a small budget, and Meera was doing it alone.

She built the system file first, which is the part that pays: hard side light from camera left, 35mm at waist level, a palette of indigo, raw cotton, brass and deep brown with no green anywhere, a crop convention with the subject in the left third, and a finish of colour negative grain with lifted blacks. Then the character sheet: a contact sheet of thirty portraits, one chosen, refined with seed variation at 0.15 until the face was specific rather than average, and seven angles built from it using a face adapter at 0.8.

The decision came at the point where it always comes. Twelve images, seen side by side, four of them printed at A3. An adapter at 0.75 gets roughly seventy per cent of an identity in thirty seconds and drifts between frames. A trained LoRA gets most of the rest and costs about an hour on free Kaggle GPU hours, plus assembling and captioning the training set, plus testing epochs — call it two hours the first time.

She ran the fifteen-minute test rather than guessing: four frames with the adapter, in the four most different situations the brief contained, laid out at delivery size, shown to her flatmate who had not been in the project. The answer was immediate. The flatmate said it looked like two different women. That settled it.

Then the co-operative's secretary made an offer that seemed helpful. One of their actual weavers, Kalaivani, has been photographed hundreds of times over the years for exhibitions and government schemes. He had a folder of

about ninety images. Train it on her, he said, and the character will look like a real person because she is one.

The technical case was strong. Ninety real photographs across many years, many lightings, many angles — the opposite of the dataset fault where every image came from one afternoon against one wall. The LoRA would almost certainly be better than one trained on a character sheet.

The objection was not technical. A LoRA of a person is a tool for producing unlimited images of that person doing things she has not done, and the folder was not consent. Kalaivani had agreed, years ago, to be photographed for an exhibition. She had not agreed to have her likeness trained into a file that a designer keeps and reuses, appearing in a festival campaign she has not seen. The photographers who took the images have rights in them too, and nobody in the room knew who several of them were.

The cost of refusing was concrete: a harder, slower route to a weaker LoRA, on a three-week deadline. The cost of accepting was a permanent asset built on an agreement nobody had, in a campaign that would be printed and posted publicly, for a co-operative whose whole standing rests on its relationship with its weavers.

YOUR ANSWERS

1. Why was the folder of ninety photographs not consent, given that Kalaivani had agreed to be photographed?

2. Meera picked epoch six rather than epoch ten. What was the symptom that decided it?

3. She showed four adapter frames to a flatmate rather than judging them herself. Why is that the better test?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 312.

CASE STUDY · MODULE 6

How it played out

Meera asked to speak to Kalaivani, which took a day to arrange and turned out to be the shortest part of the problem. Kalaivani agreed, on two conditions she raised herself: that the campaign say the images are generated, and that she see the twelve frames before they were posted. Meera wrote both into the spec, took a written note of the agreement with the secretary as witness, and asked for the thirty photographs Kalaivani could identify the photographer of. Thirty varied images is more than enough — the dataset checklist was satisfied on lighting, angle and distance, and she discarded four near-duplicates from the same exhibition burst. The LoRA trained in thirty-five minutes on Kaggle; she picked epoch six of ten, because epoch nine had started putting the exhibition hall's grey backdrop behind the character wherever she went. The campaign shipped on time. Kalaivani asked for one frame to be changed, which was done. The disclosure line cost nobody anything, and the secretary now asks for written permission as a matter of course.

The questions, discussed

1. Why was the folder of ninety photographs not consent, given that Kalaivani had agreed to be photographed?

She agreed to a photograph for a stated purpose. A LoRA is a different thing: a reusable tool that produces unlimited new images of her doing things she has not done, kept by a third party, for purposes not agreed. The photographers also hold rights in the images themselves. Permission has to be for the actual use, in writing, and is much cheaper to get before training than after a campaign has shipped.

2. Meera picked epoch six rather than epoch ten. What was the symptom that decided it?

The exhibition hall's grey backdrop began appearing behind the character regardless of the prompt. That is overfitting: the LoRA learns whatever is constant across the training set, and if a location recurs it becomes part of the

concept. Saving every epoch and comparing the same prompt and seed across them is how the middle epoch gets found — identity present, prompt still working.

3. She showed four adapter frames to a flatmate rather than judging them herself. Why is that the better test?

Because she had been staring at the character for days and could see the differences either way. The question is not whether the differences are visible to her but whether a stranger, at delivery size, in the order the audience will see them, would say it is the same person. That answer is usually immediate and usually right, and it replaces an hour of opinion with fifteen minutes of evidence.

MODULE 6 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. 413. If two go wrong, re-read the module before the exam.

1 Why will a fixed seed not give you the same character in a new pose?

A. Variation strength above 0.1 discards the identity from the noise.

B. Seeds are stored per resolution, and changing the pose changes the crop as well.

C. A seed reproduces one image, not an identity; one word changes the face.

D. Seeds only bind identity on models trained with face embeddings.

2 Twenty training photographs were all taken in one session against one wall. What will the LoRA do?

A. Produce a sharp face that ignores every prompt about clothing and hair.

B. Treat that wall and that light as part of the person's identity.

C. Overfit inside two epochs and reproduce the training images wholesale.

D. Fail to converge, because twenty images is below the usable minimum.

3 Your character LoRA always puts the subject in the same shirt. What went wrong in the captions?

A. The shirt was not captioned, so it attached to the trigger token.

B. The captions were too long and exceeded the encoder's window.

C. The trigger token appeared in only some of the captions.

D. The captions were produced by a tagger instead of written by hand.

4 You have four frames from an adapter at 0.75, side by side at delivery size. How do you decide whether to train a LoRA?

A. Count the frames where the adapter weight had to be changed.
B. Run the same four seeds through a second adapter and compare.
C. Zoom to 200 per cent and compare the features that drift the most between frames.
D. Ask a stranger whether it is the same person, at delivery size.

5 Why is a product harder to hold consistent across a set than a face?

A. Objects have harder edges, which identity adapters bind to much less reliably.
B. A product appears at more angles across a campaign than a face does.
C. Viewers compare it against the real thing, so any variation is a defect.
D. Diffusion models were trained on far more people than manufactured objects.

6 One set has a slightly drifting face and one rigorously held look. Another has a perfect face and twelve different lighting setups. Which reads as a set?

A. Neither of them; a set requires identity and treatment held to the same standard.
B. The first, because coherence is read from light, lens, palette and finish.
C. The second, because grading can be matched afterwards and identity cannot.
D. The second, because identity is what a viewer checks first.

7

MODULE 7

Finishing and handing over

Take an approved frame to a delivered file — reach the output resolution the medium genuinely requires, prepare colour for screen or press, set type that stays legible over a photographic background, run a named twelve-point QA pass, and hand over a package with disclosure, alt text and archived source files.

61 Upscaling either restores detail or invents it 318
62 Nothing goes to a client straight from the model 321
63 300 dpi is one number applied to everything 325
64 The blue on your screen is not going on the press 329
65 Making words readable on a photograph 333
66 Twelve things to check before you send 336
67 Exporting without wrecking it at the last step 340
68 Describing the image, including what it is 344
69 When "just change one thing" is a rebuild 348
Case study: Seventy megapixels nobody could see 352
Module 7 test 358

Lesson 61 · 8 min

Upscaling either restores detail or invents it

THE ONE THING TO KEEP

Diffusion-based upscalers re-generate rather than sharpen, so never point one at text, a logo, a product detail or a face you need to stay recognisable.

Three unrelated operations share one button name

Your 1024-pixel image needs to be a poster. You press Upscale. What happens next depends entirely on which of three completely different things that button does, and the tools rarely tell you.

One: resampling. Honest and useless

Bicubic and Lanczos interpolation. More pixels, no new information, mathematically averaged from what was already there. The result is bigger and softer.

It never lies to you and it never helps. Use it when you need a specific pixel dimension and the detail is already sufficient. Free in Photopea, GIMP, Krita, and every phone gallery app.

Two: super-resolution models. Conservative invention

Real-ESRGAN, SwinIR, 4x-UltraSharp, and Topaz Gigapixel on the paid side. These are networks trained on pairs of degraded and clean images, so they have learned what blur usually hides. Given a soft edge they predict a sharp one.

They do invent, but locally and conservatively. Fabric becomes plausible fabric weave. Hair becomes plausible hair. Straight lines get straight. The invention stays inside the shapes that were already there.

Free and excellent: **Upscayl**, a desktop app that wraps these models — download, drag the image in, choose a model, done. It runs on a GPU if you have one and on a CPU slowly if you do not. **chaiNNer** and ComfyUI's upscale nodes give you the same models with more control.

For faces specifically, GFPGAN and CodeFormer restore facial detail well, with a caution: CodeFormer at high fidelity settings smooths skin into plastic. Back the strength off.

Three: diffusion re-generation. Heavy invention

Ultimate SD Upscale, tiled diffusion, and anything a hosted tool labels "creative upscale" or "enhance". These cut the image into overlapping tiles, run image-to-image on each tile at a higher resolution, and stitch them back together.

The results can be spectacular. They are also **substantially a new image**. Small text becomes different text — plausible letterforms, wrong words. A distant face becomes a different face. A watch gains numerals that were never there. A pattern of tiles on a roof reorganises itself.

The mechanism behind the classic artefact: each tile is denoised knowing only its own contents plus a little overlap. A repeating pattern crossing a tile boundary has no way to stay consistent, and a large face split across two tiles gets two different skin textures. Larger tiles, more overlap, and a lower denoise (0.2-0.35) reduce it. They do not eliminate it.

The rule that follows

The more an upscaler invents, the less you can point it at anything that carries information. Text, logos, product detail, a recognisable face, a diagram, a map. For those, use resampling or a conservative super-resolution model, and put real type back over the top.

For a landscape, a texture, an abstract background — let it invent. That is where the spectacular results come from and nothing is lost.

Order of operations

Upscale last. Always.

Fix defects at the base resolution, do your colour and contrast pass, and only then enlarge. Two reasons, both practical: upscaling bakes errors in at four times the size, and if you have to re-run anything, every step after the upscale costs four times as much time and money.

The print reality nobody mentions

At 300 dpi, an A4 page is roughly 3500 × 2480 pixels. A 1024 × 1024 generation printed at 300 dpi is about 8.7 cm square — smaller than a postcard.

So anything destined for print needs real enlargement, and that means the invention question above is not optional, it is the job. For logos and anything with type, the answer is not to upscale at all but to rebuild it as vector. Vector and print covers that properly, including why a printer's grey box appears behind your transparent logo.

Large-format printing is more forgiving than people assume — a billboard is viewed from thirty metres and 30 dpi is fine. A brochure held in the hand is not forgiving at all.

Today: take one generated image and enlarge it 4× twice — once in Upscayl with a standard model, once with any "creative" upscale you have access to. Zoom to 100% on the smallest detail in each. The difference will make the rule permanent.

BEFORE YOU MOVE ON

An illustrator enlarges a poster 4× using a creative diffusion upscaler and the small print in the corner comes back as different words in the same style. What is the explanation?

A. A diffusion upscaler re-generates each tile as a new image rather than sharpening it, so it produces plausible letterforms rather than the ones that were there; anything carrying information should be enlarged with a non-generative model, or set as real type afterwards.

B. The upscale model was trained mostly on photographs, so it handles graphic content and lettering poorly.

C. The text was too small in the original; starting from a higher base resolution would have preserved it.

D. The tiles overlapped too little, so the words were split across tile boundaries.

The answer and why: p. 420

Lesson 62 · 8 min

Nothing goes to a client straight from the model

THE ONE THING TO KEEP

Curves for contrast, a colour pass to match the set, fine grain to fake a sensor, and real type over any generated words — that is the ten minutes that separates output from a deliverable.

The tell is not six fingers

You can spot a raw generation across a room, and it is rarely because of an anatomical error. It is because of five specific, fixable things — and fixing them takes about ten minutes in a free editor.

What is actually wrong with a raw generation

The tone curve is mushy. Generated images tend to sit in the midtones with weak blacks and no clean white. Open Curves, pull the bottom-left point right until the darkest shadow is genuinely black, and pull the top-right point left until something in the frame is genuinely white. Then put a gentle S in the middle. Twenty seconds, and it is the single largest improvement available.

Not Brightness/Contrast — that shifts everything uniformly and crushes both ends. Curves, because you are choosing which tones move.

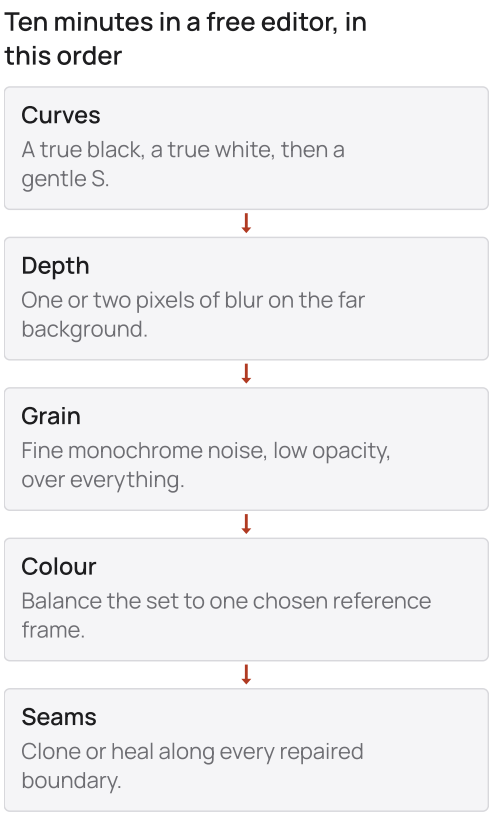
Everything is equally sharp. A real lens is sharp at the focal plane and falls off. A generated image is often uniformly crisp from foreground to horizon, which the eye reads as artificial without knowing why. Select the far background, add a small Gaussian blur — one or two pixels at 1024 wide — and the frame acquires depth.

There is no grain. Every photograph ever taken has sensor noise or film grain. Generated images have none, and the absence is conspicuous. Add a fine, uniform monochrome noise across the whole image at low opacity. This is the cheapest single move that makes an image read as photographic.

The colour is model colour. Every model has a cast. Put two generations side by side and they will not match, which is why a set of six looks like a stock-photo bundle rather than a campaign. Pick one image as the reference, then colour-balance the other five to it — usually a small shift in the shadows and highlights, per image.

Inpainted regions have soft seams. A pass with the clone or heal tool along the boundary, at low opacity.

Figure 7.1



Not Brightness and Contrast, which shifts everything uniformly and crushes both ends. Curves, because you are choosing which tones move — and it is the single largest improvement available.

Text: always replace it

If there is a word in your image, replace it with real type. Always, even from a model that renders text well.

Three reasons. Generated letterforms are subtly wrong at large size — spacing that no typeface would use, an inconsistent stroke weight on one letter. The text is not editable, so a client's copy change means regenerating the whole image. And the kerning and alignment are the parts a designer is paid for. Set the type in Photopea, GIMP, Inkscape or Canva over the generated plate. Design foundations covers how to set it well.

The free stack, and the phone

Photopea runs in a browser, opens and saves PSD files, has Curves, layers, masks and healing, and costs nothing. It is the answer for anyone on a Chromebook, a school computer or a machine that cannot install software. It works in a phone browser too, though it is a fight.

GIMP and **Krita** are free desktop applications; Krita is more comfortable for painting and masking, GIMP for general retouching. **darktable** and **RawTherapee** do the tone and colour pass better than either.

On a phone: **Snapseed** has a proper curves tool and a grain control, both free. That is genuinely enough for the tone, colour and grain pass described above. Do it on the phone, and the image will already be ahead of most raw output.

The full technique for all of this is image editing. This lesson is about knowing that you have to.

Work non-destructively, because the note is coming

Keep the generation on its own layer. Do the tone pass as an adjustment layer above it, the grain as a separate layer, the type as live text. Export flattened, keep the layered file.

The client will ask for one change. If you baked everything into the pixels, one change is a regeneration. If you worked in layers, it is a click.

Name your files with something you will understand in six months, deliver flattened PNG or JPEG at the size actually needed, and keep the working file and the original generation, seed included.

Today: take your best generated image, open Photopea, apply Curves and add fine grain. Put the before and after side by side. Ten minutes.

BEFORE YOU MOVE ON

Six generated images for one brand page each look fine alone, but together they look like an assortment from a stock library rather than one campaign. What is the most effective fix?

A. The prompts were not consistent enough across the six and should be standardised.

B. They were generated on different seeds, which is why the look drifts between them.

C. They need a style LoRA trained on the brand's existing photography before regenerating the set.

D. Every generation carries its own contrast and colour cast, so the set only reads as a set after a per-image curves and colour-balance pass matched to one chosen reference image.

The answer and why: p. 420

Lesson 63 · 8 min

300 dpi is one number applied to everything

THE ONE THING TO KEEP

Required resolution is set by viewing distance rather than by a universal figure, so a billboard needs far fewer pixels per inch than a brochure and most "high res please" requests are satisfied by a file you already have.

Where the number came from

300 dpi is the conventional figure for printed material held in the hand, and it comes from the resolving power of the eye at about 30 cm: roughly 300 distinguishable points per inch. It is a good number for a magazine page.

It is a terrible number for a billboard, and applying it everywhere is how people end up trying to produce a 30,000-pixel image for something a viewer sees from across a car park.

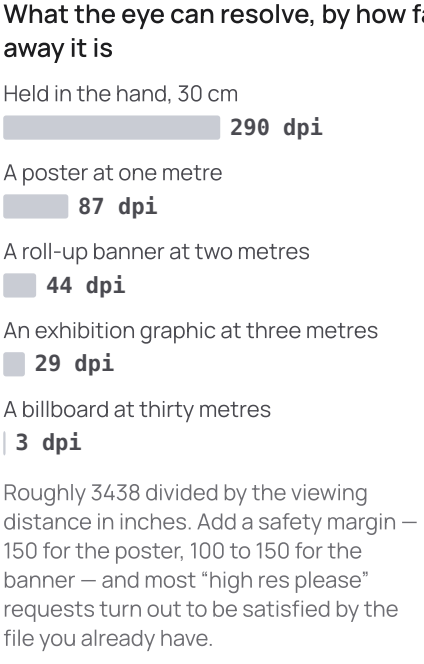
The arithmetic

The eye resolves around one arc-minute. Converted to a practical rule:

required dpi ≈ 3438 ÷ viewing distance in inches

Which gives, rounding sensibly:

Figure 7.2



- **Held in the hand, 30 cm** — 290 dpi. Use 300.
- **A poster at 1 metre** — 87 dpi. Use 150 for safety.
- **A roll-up banner at 2 metres** — 44 dpi. Use 100–150.
- **An exhibition graphic at 3 metres** — 29 dpi. Use 72–100.
- **A billboard at 30 metres** — 3 dpi. Printers commonly use 10–15.

Worked example. An A1 poster, 594 × 841 mm, viewed at a metre:

```
at 300 dpi  =  7016 x 9933 px   (70 megapixels, ~200 MB TIFF)
at 150 dpi  =  3508 x 4967 px   (17 megapixels, ~50 MB)
```

The 150 dpi version is indistinguishable at the real viewing distance and is a quarter of the work. The 300 dpi version needs a generative upscale that will invent detail nobody will see.

Ask the printer, not the internet

Commercial printers have a preferred figure for each process and will tell you in one email. The screen used for large-format printing is coarser than for litho, and supplying more resolution than the screen can carry does nothing at all.

The question to ask: "What resolution do you want artwork at, at final size, for this process?" The answer is frequently lower than you expected, and it is authoritative in a way no general advice is.

For screen

Different problem, same discipline.

- **Web hero**: 2× the layout size for high-density displays, so a 1280-pixel slot takes a 2560-pixel image. Beyond 2× there is no visible gain and a real cost in page weight.
- **Social**: the platform's stated size. Uploading larger means their compression does the resizing, usually worse than you would.
- **Email**: 600–700 pixels wide, because that is the width of most email clients.
- **Presentation slides**: 1920 × 1080 is enough for anything projected.

The web case deserves emphasis because it is where oversupply actually hurts. A 6000-pixel hero image served to a phone is several megabytes of download for a 400-pixel display, and page weight is a real cost paid by the people least able to afford it.

How this changes the workflow

If the deliverable is a roll-up banner at 100 dpi, your 1024 render upscaled conventionally to 3500 pixels is finished. No tiled diffusion, no generative upscale, no forty minutes.

So calculate first. The sequence is: final physical size, viewing distance, required dpi, required pixels, and only then decide what enlargement method the gap needs. People do it in the opposite order and upscale as far as their machine allows.

The limitation

The arithmetic gives a floor, not a ceiling, and two situations justify going above it.

The client will zoom. Not the audience, the client. They will open it at 400% on a large monitor to check the product. Supplying only what the medium needs is correct and will still produce an awkward conversation, and it is worth delivering a generous file for that reason alone.

The image will be re-used. A poster image that later becomes a brochure cover needs the brochure's resolution. If re-use is plausible, produce the higher-resolution version once while the project is live rather than reconstructing it in eight months from a file you have archived.

There is also an honest caveat about the arithmetic itself: it assumes normal vision, typical contrast and a viewer who is not looking for faults. Fine repeating detail, high-contrast type and thin lines all demand more than the formula suggests. Use it for photographic content, and give type and line work the resolution they need regardless — which is another argument for setting type separately, since vector type has no resolution at all.

BEFORE YOU MOVE ON

A roll-up banner 850 mm wide will be read from about two metres. Why is 300 dpi the wrong target?

A. Because large-format printers use a coarser screen, so resolution above the screen frequency is discarded during output.

B. Because the eye cannot resolve more than about 45 dpi at two metres, so the extra pixels are invisible to any viewer.

C. Because a file of that size would exceed what most design applications can handle reliably when placing and exporting large-format artwork.

D. Because upscaling a generated image that far requires tiled diffusion, which invents detail that will not match the rest of the artwork.

The answer and why: p. 421

Lesson 64 · 8 min

The blue on your screen is not going on the press

THE ONE THING TO KEEP

Screens emit light and presses absorb it, so a set of vivid screen colours cannot be printed at all, and the conversion has to be seen and approved before delivery rather than discovered on the printed sheet.

Two different physics

A screen makes colour by emitting red, green and blue light. Add all three and you get white. A press makes colour by putting cyan, magenta, yellow and black ink on paper that absorbs some light and reflects the rest. Add all four and you get a muddy dark brown, which is why black is printed separately.

These produce different ranges of reproducible colour, and the screen's is larger in the areas people care about most: saturated blues, vivid greens, bright oranges, anything neon.

So some colours you can see on a monitor cannot be printed. Not "printed slightly differently" — cannot be printed. The press substitutes the nearest available colour, and a vivid cyan-blue arrives as a duller, greyer blue.

What actually happens

A generated image is in sRGB. Converting it to CMYK for print maps every out-of-range colour to the nearest reachable one. The typical result:

- Saturated blues lose vibrancy and shift toward purple or grey.
- Bright greens dull noticeably.
- Vivid oranges and reds lose their glow.
- Deep blacks become lighter unless a rich black is specified.
- Subtle gradients can band.

If the image was graded to look good on a screen, this conversion can be a shock.

Soft proofing, free

You can see it before printing. GIMP has soft-proofing built in: `View > Color Management > Soft-Proofing Profile`, pick a CMYK profile, and enable the gamut warning, which marks unprintable colours in a flat grey.

Two decisions follow from what you see:

If little is flagged, proceed. Convert at export and it will be fine.

If a large area is flagged — a sky, a background, the brand colour — fix it now. Reduce saturation in that range, or re-grade the image toward colours the process can reach. Doing this yourself gives a controlled result; letting the converter do it gives whatever the algorithm decides.

Scribus does the same and is the better tool if the image is going into a layout anyway.

Practical rules

Work in RGB, convert at the end. RGB is a larger space and you keep more information through the editing. One conversion, at export.

Ask which profile. Different presses and papers use different profiles — coated and uncoated stock differ substantially. The printer will tell you, and using theirs beats guessing.

Black is not black. 100% K alone prints as a washed dark grey on large areas. A rich black — around 60C 40M 40Y 100K, though printers vary — is what you want for large solid blacks. Ask.

Colour on uncoated paper is duller. The ink sinks in. Everything is softer and lighter than on coated stock, and this surprises people more than the gamut conversion does.

Screen has its own version of this

Not a gamut problem but a consistency one. Your monitor is not calibrated, the client's is different, and phones apply their own processing. An image graded on a warm, bright laptop will look cold and dark on somebody else's cheap external display.

Two mitigations that cost nothing. Check on a second device, ideally a phone, before sending. And avoid grading to the edges — an image that depends on the last 5% of shadow detail will lose it somewhere.

If you have a colorimeter, calibrate; if not, at least set the display to a neutral factory profile rather than a vivid picture mode.

The honest position

The vector-and-print course in this catalogue covers preparation for press properly — bleed, marks, overprint, profiles, preflight. What matters here is timing: **the colour conversion is a decision made while the image**

can still be changed, not a checkbox at export.

And a limitation worth stating plainly: there is no way to make a printed image match a screen exactly, because the two are physically different media. The goal is a printed result you have seen, approved and are not surprised by. A client who says the printed poster looks duller than the screen is describing physics, and the conversation goes very differently depending on whether you showed them a soft proof beforehand.

BEFORE YOU MOVE ON

Why can a vivid cyan-blue on screen not be reproduced on a CMYK press?

A. Because CMYK uses four inks rather than three channels, so the additional black plate desaturates colours as it is applied.

B. Because screens emit light while inks absorb it, and the reachable colour range of the two processes genuinely differs.

C. Because the conversion to CMYK is performed at 8 bits per channel, which is insufficient to represent highly saturated colours without quantisation.

D. Because print colour is measured under a standard illuminant that is dimmer than a backlit display, so all printed colours appear less saturated by comparison.

The answer and why: p. 421

Lesson 65 · 8 min

Making words readable on a photograph

THE ONE THING TO KEEP

Type over a photograph fails on contrast rather than on typeface, and the fixes are all about creating a controlled surface for the words rather than about choosing better fonts.

The problem, precisely

A headline over a photograph is legible only where it has enough contrast against what is behind it. A photograph has varying tone, so a single-colour headline is high-contrast in some places and invisible in others — typically the light letters that pass over a bright cloud.

This is a measurable property, not a matter of taste. The accessibility threshold widely used for text is a contrast ratio of **4.5:1** for normal text and **3:1** for large text, where large means roughly 24px or 19px bold and up. Below those, a meaningful share of readers simply cannot read it, including anyone in bright sunlight on a phone.

Generate the surface first

The cheapest fix happens before the image exists, and the brief block set it up: prompt for quiet space where the type goes. A plain overcast sky, a wall, a dark background, a shallow depth of field that reduces the background to smooth tone.

An image generated with a place for the headline needs none of the remedies below. Most images are not, which is what the rest of this lesson is for.

The remedies, in order of preference

1. Move the type. Look for the quietest area of the frame and put the words there. Free, and usually improves the composition.

2. A gradient scrim. A soft black-to-transparent gradient over the region behind the type, at 40–70% opacity. Nearly invisible as a device and it works. This is what most well-designed posters and streaming thumbnails do.

3. Darken or lighten the whole image behind. A global exposure change to the area, which reads as a deliberate grade rather than an overlay.

4. A solid or semi-transparent panel. Explicit, very legible, and a design decision — it will look like a panel, so it should look deliberate.

5. Blur the region behind. A strong blur on the area under the type. Reads naturally as depth of field if the rest of the image supports it.

6. Outline or drop shadow on the type. Last resort. A tight, subtle shadow can work; a hard outline reads as amateur work and is the most common mistake in this whole area.

Measuring rather than guessing

Do not judge contrast by eye — the eye adapts and you have been staring at the image.

Sample the *brightest* pixel behind light type and the *darkest* behind dark type; those are where it fails. Any free web contrast checker takes two hex values and returns the ratio. If the worst point clears 4.5:1, the headline clears everywhere.

For type over a photograph, this takes thirty seconds and settles an argument that otherwise runs on opinion.

Choices that help

- **Weight over size.** A medium or bold weight is far more legible over a busy background than a light weight at a larger size.

- **Avoid thin and high-contrast typefaces** over photographs. A Didone with hairline strokes disappears into any texture.
- **Fewer words.** A six-word headline can be set large enough to survive. A sentence cannot.
- **Test at the delivered size.** Type that reads on your monitor at 100% may be unreadable as a 400-pixel feed thumbnail.

Where the type is set

Not in the generator, as the repair block established. Inkscape for anything going to print, GIMP or Krita for raster work, Scribus for multi-page layout, Photopea in a browser. All free, all with proper typographic control over tracking, leading and alignment.

Deliver the layered file as well as the flattened image so the headline can be changed later without regenerating anything.

The judgement this does not replace

Contrast ratios tell you whether words can be *read*. They say nothing about whether the type belongs to the picture — whether its weight suits the image's weight, whether it sits where the composition wants it, whether the whole thing is any good.

That is design, and the design-foundations course in this catalogue covers hierarchy, spacing and alignment properly. What this lesson gives you is the floor: a headline that clears 4.5:1 at its worst point is legible, and one that does not is broken regardless of how attractive it looks on your screen in a dim room.

BEFORE YOU MOVE ON

Light type over a photograph is legible across most of the frame but disappears over a bright cloud. What is the right first response?

A. Add a hard outline to the type so that it retains a defined edge against any background it passes over.

B. Increase the type size, since larger text clears the 3:1 threshold rather than the 4.5:1 one and is therefore easier to make legible.

C. Move the type to the quietest part of the frame, or place a soft gradient scrim behind it.

D. Convert the type to a darker colour, which will contrast against the bright cloud rather than disappearing into it.

The answer and why: p. 422

Lesson 66 · 8 min

Twelve things to check before you send

THE ONE THING TO KEEP

A written check performed at 100% on the final file catches the specific faults that generated images produce and that the person who made the image has stopped being able to see.

Why a list rather than looking

You have been looking at this image for three hours. You cannot see it any more — your eye skips the areas you have already resolved, and the fault you missed at hour one is now invisible to you permanently.

A written list defeats this, because it directs attention rather than relying on it. It is the same reason surgeons and pilots use checklists, and the same reason it feels unnecessary right up until it catches something.

The twelve

Run these at 100% on the final flattened file, not on the layered working document.

1. Hands and fingers. Count them. Every visible hand.

2. Eyes. Both looking the same way, matched in size, catchlights consistent with the light source.

3. Teeth. Even but not uniform, correct in number, no extra row.

4. Ears, glasses, jewellery. Glasses arms that reach the ears, earrings matching in pairs, watches with plausible faces, necklaces that pass behind rather than through.

5. Text. Anything legible in frame. Signage, labels, book spines, screens. Either correct or deliberately illegible, and never accidentally spelling something.

6. Logos and brand marks. Against the forbidden list. Including background signage, clothing and packaging.

7. Shadows. Every one pointing consistently. Every object that should cast one having one.

8. Reflections. Mirrors, windows, glass, polished surfaces, water. Reflecting something that is actually in the scene.

9. Repeating patterns. Tiles, brickwork, fabric, foliage, crowds. Looking for the repeat, the discontinuity, and the face that has been duplicated in a crowd.

10. Edges and seams. Around every composited element, every inpainted region, every outpainted extension.

11. Anatomy and structure. Limbs attached correctly, furniture with the right number of legs, walls meeting floors, stairs that go somewhere.

12. Banding and compression. Gradients and skies, viewed at 100% on a good screen.

Then the three views

At delivery size. How the audience sees it. Faults invisible here are not faults.

As a thumbnail, 200 pixels. Does it still read? Is the subject clear?

On a phone. Different gamma, different brightness, often different colour handling. Send it to yourself and look.

The second pair of eyes

The single most effective step, and the cheapest. Show it to somebody who has not been working on it, for ten seconds, and ask what they notice.

They will find the thing you cannot see. Every time. Ten seconds is enough — you want a first impression, not a considered review, because a first impression is what your audience will have.

If there is nobody to ask, the next best thing is time. Look at it tomorrow morning. Failing that, flip it horizontally, which resets your eye's adaptation and makes compositional and structural faults jump out. This is an old illustrator's trick and it genuinely works.

Recording the pass

For client work, keep the record:

```
spring26-07-final-v03   QA 18 Mar
  hands ok / eyes ok / teeth n-a / jewellery: bangle fixed v03
  text: none in frame / logos: none — checked background signage
  shadows ok / reflections: window ok / patterns: tiles ok
  edges: jar composite re-feathered / anatomy ok / banding: sky
ok
  delivery size ok / thumbnail ok / phone ok
  second pair of eyes: R, no notes
```

Two minutes. It means that if something does surface later you know whether it was checked, and it makes the pass an actual event rather than an intention.

What the list cannot do

It catches faults. It does not tell you whether the image is any good, whether it answers the brief, or whether it will work in its placement. Those are judgements made much earlier, and a technically flawless image that misses the brief has failed completely.

The list also has a real failure mode worth naming: applied mechanically, it produces images that are correct and lifeless, because everything unusual has been checked out of them. Some asymmetry, some imperfection and some unresolved detail is what makes a photograph look photographed. The list exists to catch things that are *wrong*, not things that are merely irregular, and telling those apart is judgement that no checklist supplies.

BEFORE YOU MOVE ON

Why does flipping an image horizontally help at the end of a long session?

A. Because it reveals composition faults that only appear when the visual weight of the frame is reversed against the reading direction.

B. Because your eye has adapted to the familiar version, and reversing it breaks that adaptation so structural faults become visible again.

C. Because generated images frequently contain a directional bias inherited from training data, which a flip makes apparent.

D. Because inpainted and composited regions were worked on in one orientation, so their seams are easier to detect when the image is viewed the other way round.

The answer and why: p. 422

Lesson 67 · 8 min

Exporting without wrecking it at the last step

THE ONE THING TO KEEP

Export is where visible damage is introduced by compression, colour profile loss and resampling, and each has a specific setting that prevents it.

Losing it at the last step

Three hours of work, and the file you send has banded skies, soft edges and a colour cast. All three are export faults, all three are avoidable, and all three are common.

Formats, and what each is for

JPEG — photographic content, no transparency. Quality 85–92 for delivery; below 80 shows artefacts in gradients and at edges. Progressive encoding for web, because it renders in passes as it loads.

PNG — transparency, flat colour, anything with type or sharp edges. Lossless, therefore large. A PNG of a photograph is often 5–10× the equivalent JPEG for no visible gain.

WebP — the sensible web default. Smaller than JPEG at the same quality, supports transparency, supported everywhere that matters now. Quality 80–85.

TIFF — print delivery and archiving. Lossless, 16-bit capable, carries CMYK and profiles. The format to hand a printer.

PDF — anything typeset. Vector type stays vector, so it is sharp at any size. For press, ask which PDF/X standard they want.

The commonest mistake is delivering a JPEG for print. Ask for TIFF or PDF, because the compression artefacts invisible on a screen are visible on paper, particularly in skies and skin.

Figure 7.3

Two destinations, and the two settings that decide the outcome

	Format	Profile
For a screen	WebP at 80–85 JPEG 85–92 also fine	sRGB, embedded or colours shift
For a press	TIFF or PDF never a JPEG	The printer's ask which one

A file without an embedded profile is interpreted by whatever opens it, which causes more "it looks different on my computer" than anything else in this course.

The settings that matter

Colour profile: embed it. A file without an embedded profile is interpreted by whatever opens it, and the usual symptom is an image that looks washed out or oversaturated on someone else's machine. Export sRGB with the profile embedded for screen, and the printer's CMYK profile embedded for print. This single setting causes more "it looks different on my computer" than anything else.

Bit depth. 8-bit for delivery. 16-bit for anything that will be edited further — the extra depth is what prevents banding when a gradient is adjusted again.

Resampling method. If the export resizes, the method matters. Lanczos for downscaling; bicubic is acceptable. Nearest neighbour is for pixel art only, and using it by accident produces a harsh, aliased result.

Metadata. Decide deliberately. Camera and software metadata is usually harmless; generation prompts, which some interfaces embed, may not be something you want to ship to a client or publish. Strip it for delivery, keep it on your archived original.

Banding, and how to prevent it

Banding — visible steps in a gradient, most often in skies — comes from too few distinct values to describe a smooth transition, made worse by compression.

Three preventions:

1. **Work in 16-bit** when the image has large smooth gradients.
2. **Add a small amount of noise** before export. Two or three per cent of monochrome noise breaks the bands up and is invisible. This is why the grain step in the composite lesson helps here too.
3. **Export at higher JPEG quality**, or use PNG or WebP for images dominated by gradients.

Check the exported file

Not the working document. Open the actual delivered file and look at it, because export is a transformation and transformations have faults.

Specifically: skies and gradients at 100%, edges of composited elements, the darkest and lightest areas for clipping, and the overall colour against the working file.

Thirty seconds, and it catches the export that silently converted to a different profile.

Sizes to deliver

Give the client what they need and nothing confusing:

```
deliver/
  spring26-07-print-A3-300dpi.tif       CMYK, printer's profile
  spring26-07-web-2560.webp             sRGB
  spring26-07-social-1080x1350.jpg      sRGB
  spring26-07-thumb-600.jpg             sRGB
  spring26-07-layered.xcf               editable, type live
  README.txt                            what each file is, and
the disclosure note
```

The README is the part people skip and the part clients appreciate most. Four lines saying which file is for what prevents the print file being uploaded to Instagram and the social crop being sent to the printer, both of which happen regularly.

The limitation

None of this recovers quality that was lost earlier. An image upscaled badly, over-sharpened or graded into clipped blacks cannot be exported well; export preserves what it is given.

And a note on platform compression: social platforms re-encode whatever you upload, often aggressively, and your carefully exported file is not what viewers see. You can improve the outcome slightly by uploading at exactly the platform's stated dimensions so it does not resample, and by avoiding fine high-contrast detail that compression handles badly. You cannot prevent it, and an image that depends on subtle gradient detail will be damaged by a system you have no access to.

BEFORE YOU MOVE ON

An exported JPEG looks washed out on the client's machine but correct on yours. What is the most likely cause?

A. The client's monitor is uncalibrated, which is the usual reason two people see the same file differently.

B. The JPEG quality was set too low, and the compression has flattened the tonal range across the whole image.

C. The colour profile was not embedded, so the receiving application interpreted the values with a different profile.

D. The file was exported at 8 bits per channel rather than 16, losing the tonal precision that held the image's contrast together.

The answer and why: p. 423

Lesson 68 · 8 min

Describing the image, including what it is

THE ONE THING TO KEEP

Alt text describes what an image conveys in its context rather than listing what is in it, and whether to mention that an image is generated depends on whether that fact matters to understanding it.

Who it is for and what it does

Alt text is read aloud by screen readers, shown when an image fails to load, and used by search engines. It is the image's content for anybody who is not seeing it, and it is a normal part of delivering an image rather than an optional extra.

It is also frequently done badly, in a way that makes things worse rather than better.

The rule that decides the content

Describe the function, not the inventory.

Ask what the image is doing in its context, and write that. The same photograph gets different alt text depending on where it appears.

- On a product page: `A terracotta jar of mango pickle, lid removed, on a wooden table.`
- In an article about a family business: `A woman in her sixties sealing jars of pickle in a courtyard.`
- As a decorative background behind a headline: `alt=""` — empty, so screen readers skip it.

That last case is the one people get wrong most often. Alt text on a purely decorative image is noise: the reader hears a description of something irrelevant between them and the content. An empty alt attribute is the correct, accessible answer for decoration.

Writing it

- **Roughly 125 characters.** Some screen readers truncate around there, and brevity is a kindness.
- **Do not start with "Image of" or "Picture of".** The screen reader has already said it is an image.
- **Lead with what matters.** The subject first, context after.
- **Include text that appears in the image.** If a sign or a headline is part of the picture, it must be in the alt text or that information is unavailable.
- **Do not stuff keywords.** It degrades the experience for the people the attribute exists for, which is the whole point of it.

Should you say it is AI-generated?

Genuinely contested, and the honest answer is that it depends on whether the fact is relevant to understanding the image.

The case for including it: a person using a screen reader should have access to the same information a sighted person has. If the image is visibly stylised or labelled as generated on the page, withholding it from alt text creates an inequality.

The case against: alt text is for conveying content, and a provenance note consumes characters that could describe the picture. Provenance is better carried by a caption, a credit line or embedded metadata, all of which reach everyone.

The practical position most publishers have settled on:

- **Put provenance in the caption or credit**, where everybody gets it.
- **Keep alt text for content**, unless the fact that it is generated is part of what the image is communicating.
- **Do include it** where the image might be taken as documentary evidence — a news context, a person, a place, an event.

Whatever you choose, be consistent within a project, and write it down in the spec so the client's team follows the same rule after you have gone.

The rest of accessibility

Alt text is one part. The others belong to the image itself.

Contrast, covered in the type lesson. 4.5:1 for normal text.

Do not put essential information only in an image. A price, an address, a date in a graphic is invisible to a screen reader and unsearchable. It belongs in the page text as well.

Avoid rapid flashing in anything animated — three flashes a second is the seizure threshold.

Do not rely on colour alone to convey meaning, since some readers will not distinguish the colours you chose.

The limitation

Alt text cannot carry what a complex image conveys. A detailed illustration, a chart, a photograph whose meaning is in its atmosphere — 125 characters will not do it, and pretending otherwise is a box-ticking exercise.

For those, the right answer is a longer description elsewhere on the page, or a caption doing real work, or text content that makes the image supplementary rather than load-bearing. An image carrying essential meaning that cannot be described is an accessibility problem in the design, not in the alt attribute, and no amount of careful writing in that field fixes it.

BEFORE YOU MOVE ON

An image sits behind a headline purely as decoration. What is the correct alt text?

A. A brief description of the photograph, since every image on a page should carry a description for screen reader users.

B. A description of the mood the image conveys, so that non-sighted users receive the same emotional context as sighted ones.

C. An empty alt attribute, so screen readers skip it and do not interrupt the content with irrelevant description.

D. A note that the image is decorative, so the reader understands why no description has been provided for it.

The answer and why: p. 423

Lesson 69 · 8 min

When "just change one thing" is a rebuild

THE ONE THING TO KEEP

Revisions divide into adjustments, regional changes and structural changes, and the professional skill is saying which category a request falls into before agreeing to it.

Three categories, very different costs

Adjustment. Colour, contrast, crop, grade, grain. Minutes, and non-destructive if your files are layered.

"Warmer." "A bit brighter." "Crop tighter." "Less grain."

Regional. One area regenerated or replaced. Thirty to ninety minutes, and it may disturb the areas around it.

"Different jar." "Change her top." "Remove the object on the left." "Different expression."

Structural. The composition, the pose, the perspective, the light. This is a new image, and everything downstream — repair, compositing, grade — is repeated.

"Can she face the other way?" "Can we see more of the room?" "Can the light come from the right?" "Can there be two people?"

A client cannot tell these apart, and there is no reason they should. Saying which one a request is, before starting, is your job.

The sentence to have ready

> *"That's a structural change rather than an adjustment — it means a new frame and about half a day, because the repair and compositing have to*

> *be redone on top of it. I can do it; I want to flag it so you can decide whether it's worth it."*

Three things make it work. It is not a refusal. It states the cost in time rather than in blame. And it hands the decision to them, which is where it belongs.

Said before starting, that is competence. Said afterwards with an invoice attached, it is a dispute.

Reducing how often it happens

Approve the anchor before building the set. The consistency block's central discipline, and the largest single reducer of structural revisions.

Show composition early, cheaply. A rough contact sheet at concept stage catches "she should be facing the other way" when it costs a minute, rather than after three hours of repair.

Write the rounds into the spec. Two rounds included, structural changes are a new round. Agreed on day one, this is an administrative fact rather than an argument.

Keep the layered file. Most adjustments and some regional changes are minutes *if* the work is layered and the raw generation is preserved. If you flattened everything, an adjustment becomes a rebuild for reasons that are entirely your own fault.

When the request is unclear

"Can you make it pop?" and "it's not quite there" are real and frequent. They are not unreasonable — clients are describing a feeling and do not have the vocabulary.

Turn it into a choice. Produce three versions: more contrast, warmer, tighter crop. Send all three. They will pick one, and from the pick you learn what they meant far more reliably than from any amount of questioning.

The visual-storytelling course in this catalogue covers this conversation properly. The version that belongs here: **never guess twice.** If one attempt at interpreting a vague note misses, stop and make it concrete before spending another hour.

The one to push back on

Occasionally the request makes the work worse, and you can see it. A crop that breaks the composition, a colour that fights the brand, type that will not be legible.

Say it once, clearly, with the reason and the consequence — "at that crop the headline drops below the contrast threshold and will be unreadable on a phone" — and then do what they ask if they still want it. You are an adviser rather than a gatekeeper, and having said it once in writing is both the professional obligation and the protection.

What is not worth doing is refusing, or complying silently and resenting it. The first ends relationships and the second ends them more slowly.

The limit

Some revisions cannot be done at any price, and it is better to say so immediately. "Can we use this same image but with a different product?" on an image where the product is lit into the scene is not a revision; it is a new job. "Can you make it look like a photograph?" when the client means a real photograph of a real thing is a brief change, not a note.

Being clear about those quickly, with an alternative offered, is far better received than a week of attempts followed by an apology. Clients remember what you delivered and how predictable you were. Almost nobody remembers which revision round something was fixed in.

BEFORE YOU MOVE ON

A client asks for the subject to face the other way. How should this be categorised and handled?

A. As a regional change, since only the subject is altered while the background and lighting of the frame remain untouched.

B. As an adjustment, because a horizontal flip achieves it in seconds without any regeneration being required at all.

C. As a structural change: flag the cost before starting, since it is a new frame with all the repair and compositing repeated.

D. As a structural change that should be declined, because it invalidates the approved composition the client already signed off on.

The answer and why: p. 424

CASE STUDY · MODULE 7

Seventy megapixels nobody could see

An illustrative case, built from documented patterns.

A district administration in Ranchi printing an A1 public-health poster for four hundred and twenty panchayat noticeboards.

The artwork was finished and approved: a generated illustration of a vaccination queue at a rural health sub-centre, brand colours from the state health department, a headline in Hindi set in Inkscape over a quiet upper third. The press date was in five days. The department's procurement note said, as such notes almost always do, *artwork at 300 dpi*.

A1 is 594 by 841 millimetres. At 300 dpi that is 7016 by 9933 pixels — about seventy megapixels, and a TIFF of roughly two hundred megabytes. The base render was 1216 by 832, upscaled once conservatively to about 3500 pixels on the long edge during finishing.

Getting from there to seventy megapixels is not a resize. It is a generative upscale: tiled diffusion at low denoise, forty minutes of processing, and then a careful check of every tile boundary. Ashish, who had built the poster, knew what that would do. The illustration contains a row of small figures in the queue, a signboard on the sub-centre wall, and the department's logo composited in at the bottom. A generative upscale invents plausible detail at the scale it is asked for. Small faces become different faces. The signboard's lettering becomes different lettering that still looks like lettering. The logo, if it went through, would come back subtly not the logo.

So his choices were three, and the deadline made them unequal.

Do the 300 dpi upscale. Cost: most of a day, a mandatory mask around the logo and the signboard so they do not go through the tiled pass, a repair pass on the queue faces afterwards, and a file large enough that the department's email would reject it.

Deliver at 3500 pixels and say nothing. Cost: a procurement note says 300 dpi, and supplying something that does not meet a written spec without flagging it is how a job gets rejected at the printer by a person who is only

checking numbers.

Ask the printer. Cost: a phone call, and the risk of an answer he did not want.

The arithmetic said the third was right before the call was made. A noticeboard poster is read from about a metre, sometimes two. The rule is roughly 3438 divided by the viewing distance in inches: at one metre, 87 dpi; at two, 44. Even with a generous safety margin at 150 dpi, A1 needs 3508 by 4967 pixels — seventeen megapixels, which is a conventional enlargement of the file he already had, with nothing invented.

There was a second problem waiting, and it was the one that would actually have caused a reprint. The state health department's palette includes a vivid cyan-leaning blue, and Ashish had graded the whole illustration warm around it on a bright laptop. He soft-proofed in GIMP against the printer's CMYK profile with the gamut warning on, and a large area of the sky flagged flat grey — unprintable. Converted at export without looking, that sky would have arrived on four hundred and twenty posters as a duller, greyer, slightly purple blue, and the department would have been right to say it was not their colour.

YOUR ANSWERS

1. Why would a generative upscale have been the wrong tool here even if there had been time for it?

2. Where did the 300 dpi figure come from, and why is applying it to a poster the wrong move?

3. The colour problem would have caused a reprint. What made it visible before the press run rather than after?

STOP HERE

Write your answers before you turn the page. How it played out starts on p. 356.

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

CASE STUDY · MODULE 7

How it played out

The printer's answer to the direct question — what resolution do you want artwork at, at final size, for this process — was 150 dpi, and he volunteered that the screen for large-format work could not carry more anyway. Ashish delivered a 3508 by 4967 TIFF, produced by a conservative super-resolution model in about four minutes, with the logo and the signboard composited back in at full resolution after the enlargement rather than put through it. He fixed the gamut problem before converting, by desaturating the sky range and re-grading the illustration toward colours the process could reach, and he printed one A4 section at the printer's shop as a physical proof and took it to the department before the run. The health officer approved the proof in the room. Total time saved against the day the 300 dpi route would have cost: about six hours, all of which went into the proof and the colour work, which is where it mattered. The procurement template now says *at final size, at the resolution confirmed by the printer for this process*.

The questions, discussed

1. Why would a generative upscale have been the wrong tool here even if there had been time for it?

Because it invents rather than restores. The queue's small faces, the signboard lettering and the logo all carry information, and a tiled diffusion pass would have produced plausible replacements for them — different faces, letterforms that look like letters and are not, a logo that is subtly not the mark. The rule is that anything which must be correct does not go through a generative upscale; it is composited back in at full resolution afterwards.

2. Where did the 300 dpi figure come from, and why is applying it to a poster the wrong move?

It comes from the resolving power of the eye at about thirty centimetres, which is how far away a magazine page is held. Required resolution is set by viewing distance — roughly 3438 divided by the distance in inches — so a

poster read at a metre needs about 87 dpi and 150 with a safety margin. Applying the hand-held number everywhere is how people generate files nobody can see the detail in.

3. The colour problem would have caused a reprint. What made it visible before the press run rather than after?

Soft proofing against the printer's own CMYK profile with the gamut warning on, before converting. A large flagged area is a decision point: re-grade toward colours the process can reach, under control, rather than letting the converter choose a substitute. A printed proof approved in the room then turns "it looks duller than the screen" from a complaint into a thing everybody had already seen.

MODULE 7 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. 419. If two go wrong, re-read the module before the exam.

1 Which image should never go through a diffusion-based "creative" upscale?

A. A packaging shot with a legible product label.
B. A generated sky with a smooth gradient.
C. A landscape with foliage, distant hills and no text anywhere in it.
D. An abstract texture used as a background.

2 Why Curves rather than Brightness and Contrast on a raw generation?

A. Brightness and Contrast is applied destructively in every free editor available.
B. Curves is the only tool that can set a true black point in sRGB.
C. Curves works in 16 bits and Brightness and Contrast does not.
D. Curves lets you choose which tones move instead of shifting all of them.

3 An A1 poster will be read from about a metre. What resolution does it need?

A. 300 dpi, because that is the standard for any printed material.
B. About 87 dpi by the arithmetic, so 150 with a safety margin.
C. 600 dpi, because large-format screens are finer than litho screens.
D. 72 dpi, because print and screen use the same pixel density.

4 A vivid cyan-blue sky flags grey under a CMYK soft proof. What should you do, and why then?

A. Convert at export and let the profile pick the nearest reachable colour.

B. Raise the saturation so more of the range survives the conversion.

C. Re-grade that range yourself now, while the image can still be changed.

D. Deliver in sRGB and let the printer handle the conversion at their end.

5 A headline sits over a photograph. Where do you sample to test whether it is legible?

A. Across an average of the whole area behind the headline.

B. At the darkest point behind light type, which is the worst case.

C. At the centre of the type block, where most of the letters fall.

D. At the brightest point behind light type, where it fails.

6 A client says the delivered file looks washed out on their machine. Which export setting is the likeliest cause?

A. No colour profile was embedded, so the file was interpreted.

B. JPEG quality was set at 85 rather than at the maximum the encoder allows.

C. The bit depth was 8-bit rather than 16-bit for delivery.

D. The resampling used bicubic instead of Lanczos on downscale.

EXAMINATION

Image Generation, in Practice — final examination

This paper covers the whole course:

- assembling a working setup and knowing what it costs you
- turning a request into a written specification
- exploring to a frame worth keeping
- steering with images rather than adjectives
- repairing and extending and compositing
- holding a character and a product across a set
- finishing a file for the medium it is going to

56 questions, the whole pool. The site draws 25 for a sitting, pass mark 70%. Work any 25 under your own conditions. Answers from page 425.

1 Module 1 · The kit, and what each piece of it costs you

A 16 GB M2 MacBook against an 8 GB Nvidia desktop card, for SDXL work. What is the honest comparison?

A. The Mac is three to five times slower but can load models the 8 GB card cannot.

B. The Mac is faster, because unified memory removes the transfer over the PCIe bus entirely.

C. They are equivalent, since Metal and CUDA compile to the same kernels.

D. The Mac cannot run these models at all, because they require CUDA.

2 Module 1 · The kit, and what each piece of it costs you

What is the catch shared by Colab's free tier, Kaggle notebooks and Hugging Face Spaces for image work?

A. They cap output resolution at 512 pixels on the free tier.

B. They watermark every image that is generated on their free hardware.

C. They block diffusion models outright under their published acceptable-use policies.

D. Nothing persists, so you reinstall and re-download the model every session.

3 Module 1 · The kit, and what each piece of it costs you

Why does the course insist on .safetensors rather than .ckpt?

A. safetensors files are smaller, so they load faster from disk.

B. A .ckpt is a Python pickle, so loading it executes code from the file.

C. The .ckpt format cannot store fp8 or quantised weights.

D. Only safetensors files carry the model hash needed for reproducibility.

4 Module 1 · The kit, and what each piece of it costs you

You will run the same eleven-step process over image after image for a paying client. Which interface does the course recommend, and why?

A. Fooocus, because good defaults remove the eleven steps entirely.

B. Forge, because a settings panel is the fastest to learn and the tutorials match.

C. ComfyUI, because a saved graph does not decay the way notes do.

D. InvokeAI, because a unified canvas makes every repair one gesture.

5 Module 1 · The kit, and what each piece of it costs you

On Windows with an Nvidia card, speed collapses after you load a second model and nothing errors. What is it?

A. The second model overwrote the first and is being reloaded per step.

B. Windows paged the model out because the file was sitting on a network drive.

C. The scheduler silently fell back to a CPU implementation of attention.

D. The driver is spilling VRAM into system RAM instead of raising an error.

6 Module 1 · The kit, and what each piece of it costs you

On an 8 GB card, in what order should you attack the four consumers of memory?

A. Offload the text encoder, tile the decode, cut the batch, then quantise.

B. Reduce resolution first, because memory scales with pixel count.

C. Enable sequential CPU offload immediately, because it makes almost anything fit.

D. Quantise the weights first, since the network is the bulk of it.

7 Module 1 · The kit, and what each piece of it costs you

How do you actually test whether a Q4 model costs you anything on your own work?

A. Generate twenty images at each precision and compare the two averages side by side.
B. Read the benchmark tables on the model's community page.
C. Fix the seed, render at both, and compare small text and distant faces.
D. Compare file sizes, since quality tracks the bits per weight.

8 Module 1 · The kit, and what each piece of it costs you

In a well-run job, where do nine out of ten renders happen, and why does it matter?

A. During upscaling, which is where the credits are actually spent.
B. At low resolution and low steps, where cost per render is lowest.
C. In inpainting, because each defect needs several attempts to clear.
D. At full quality, because compositional faults hide at low resolution.

9 Module 2 · From a request to a specification you can generate against

A 1280 × 720 thumbnail is usually seen at about 246 × 138. What follows for the design?

A. Detail below a certain scale is wasted and contrast carries the frame.
B. Upscale to 2560 wide so it survives the platform's re-encoding.
C. Use a 16:9 bucket and accept the duplicated subjects it produces.
D. Generate at 246 × 138 so that no pixels are wasted on detail nobody will see.

10 Module 2 · From a request to a specification you can generate against

A client needs a product cut-out for a shop listing, on transparency. What do you tell them?

A. Transparency requires a vector model such as Recraft rather than a raster one.

B. Generative fill can produce transparency, but only inside Photoshop.

C. Any model will export a PNG with an alpha channel if you ask for one.

D. No diffusion model outputs transparency; generate on plain and cut out.

11 Module 2 · From a request to a specification you can generate against

Why is the anti-reference worth more than two of the other five slots in a reference pack?

A. It prevents the model from reproducing a competitor's visual identity.

B. It doubles as a negative prompt the model can be conditioned on.

C. It converts taste into a statement both sides can check later.

D. It is the only slot a client can supply without any art direction.

12 Module 2 · From a request to a specification you can generate against

The hardest item on a forbidden list to catch is stereotype. Why can no prompt fix it reliably?

A. Negative prompts cannot express a demographic concept in a form the model reads.

B. The model produces the average its training data associates with your words.

C. Safety filters remove the terms that would let you steer away from it.

D. Stereotype only appears after an upscale, past the point of prompting.

13 Module 2 · From a request to a specification you can generate against

You set the exact brand red in an editor and it still looks wrong in the frame. What is happening?

A. Simultaneous contrast: the surrounding light shifts how the red reads.
B. The model re-graded that region during the final denoising steps.
C. The monitor is uncalibrated, so the value on disk is not the value shown.
D. The export stripped the colour profile and shifted the hue.

14 Module 2 · From a request to a specification you can generate against

Which of these is the strongest candidate for photographing rather than generating?

A. An abstract texture for a section divider.
B. A 1970s office interior for a concept spread.
C. A mountain road at dawn behind a product.
D. The dish a restaurant is actually selling.

15 Module 2 · From a request to a specification you can generate against

Which request is a structural change rather than an adjustment?

A. Can we make it warmer?
B. Can she face the other way?
C. Can we crop it a little tighter?
D. Can we take some of the grain out?

16 Module 2 · From a request to a specification you can generate against

What does a vendor's training-data indemnity actually cover?

A. The cost of re-creating assets if the model version is retired.
B. Any claim arising from an image you generated on their service.
C. Claims about what the model learned from, not what you prompted.
D. Your ownership of the output in every jurisdiction they operate in.

17 Module 3 · Getting to a frame worth keeping

The image is framed too tightly. Which slot do you change?

A. Finish, because crop is part of the film stock.
B. Medium, because a photograph frames tighter than a painting.
C. Subject, by describing the person more fully.
D. Camera, because framing is a camera decision.

18 Module 3 · Getting to a frame worth keeping

How does prompt order differ between a CLIP-based model and a T5-based one?

A. CLIP weights earlier tokens more; T5 is far less position-sensitive.
B. Neither is position-sensitive; order has never mattered on any model.
C. CLIP handles full sentences better and T5 prefers comma-separated tags.
D. T5 models weight the last tokens most, so put the subject at the end.

19 Module 3 · Getting to a frame worth keeping

What happens when you stack five camera terms that all point in the same direction?

A. The terms conflict, so the sampler picks one of them at random on every step.
B. The token window overflows and the last two terms are dropped.
C. The distribution narrows to the most predictable image available.
D. The model averages them and produces a mid-range focal length.

20 Module 3 · Getting to a frame worth keeping

Why does naming the light also fix composition and mood as a side effect?

A. Lighting terms are weighted more heavily than composition terms.
B. Light and composition are not separable; a phrase decides both.
C. The lighting slot is read last, so it overrides the earlier slots.
D. Shadow placement is computed geometrically once a source is named.

21 Module 3 · Getting to a frame worth keeping

What can a CLIP interrogator not do, and why?

A. It cannot recover the picture, because it describes rather than inverts.

B. It cannot read non-Latin text, because its vocabulary is English only.

C. It cannot run without a GPU, so it is unavailable on a free tier.

D. It cannot describe lighting, because light is not in its training captions.

22 Module 3 · Getting to a frame worth keeping

Your portrait's composition is right and the face is nearly there. What is the cheapest next move?

A. Raise the step count from 25 to 60 for more finish.

B. Add "beautiful detailed face" to the front of the prompt.

C. Regenerate with a new seed until the face lands.

D. Four variations at strength 0.15 around the same seed.

23 Module 3 · Getting to a frame worth keeping

You measured your checkpoint's CFG burn point at 8, then changed checkpoint. What now?

A. Ignore CFG entirely, since modern samplers normalise guidance.

B. The number transfers, because CFG is defined by the sampler rather than the model.

C. Re-test, because the burn point is a property of the checkpoint.

D. Halve it, since a newer checkpoint tolerates less guidance.

24 Module 3 · Getting to a frame worth keeping

Why take two frames forward from a contact sheet rather than one?

A. A batch of two is cheaper per image than two separate requests would be.

B. Some frames unravel under repair and you need somewhere to go.

C. The client should always be shown at least two options.

D. Two seeds average into a more stable result when blended.

25 Module 4 · Steering with pictures instead of adjectives

A tool gives you one box marked "reference image". Why does it matter which of three things it is doing?

A. Structure, style and character references do completely different jobs.

B. Because a reference disables the text prompt on most implementations.

C. Because only a character reference may be used commercially.

D. Because that box only accepts images whose ratio matches the output size.

26 Module 4 · Steering with pictures instead of adjectives

Why does a 400-pixel forward from a messaging app make a poor img2img reference?

A. Messaging apps strip the metadata the model needs to read.

B. Low-resolution inputs force the denoise strength above 0.75.

C. The file is too small for the model's minimum input size.

D. Compression blocks are reproduced faithfully as texture.

27 Module 4 · Steering with pictures instead of adjectives

Somebody spends forty minutes raising denoise to move a subject to the right. What have they missed?

A. Denoise above 0.9 is disabled on most interfaces for safety.

B. By the time it can move, nothing else survives either.

C. Moving a subject needs a higher CFG rather than a higher denoise.

D. The reference should have been cropped before raising the strength.

28 Module 4 · Steering with pictures instead of adjectives

Canny conditioning on a compressed photograph gives thin phantom lines in the sky. Why, and what do you do?

A. The control weight is too low; raise it so the real edges dominate.

B. The photograph is too dark; raise the exposure before running the preprocessor.

C. Compression blocks look like edges; raise the thresholds or use softedge.

D. Canny needs a square input; crop the photograph before conditioning.

29 Module 4 · Steering with pictures instead of adjectives

A sketch-conditioned interior feels subtly impossible to everyone who sees it. What did the sketch not fix?

A. The palette, which needs a colour-blocking pass at low denoise instead.

B. The materials, which a depth conditioning pass would have supplied instead.

C. The lighting, which a scribble control has no way of carrying through.

D. The perspective, which the model followed faithfully including the error.

30 Module 4 · Steering with pictures instead of adjectives

Your depth control is weak because a distant wall is visible through a window. What is the fix?

A. Move the far clipping plane in so the range covers the scene.

B. Switch to normal conditioning, which is not normalised.

C. Render the depth pass at a higher resolution.

D. Raise the control strength until the near objects start to register properly.

31 Module 4 · Steering with pictures instead of adjectives

Two people shaking hands, each in distinct clothing. Why will regional prompting not solve it?

A. Attribute binding is solved by weighting syntax rather than by regions.
B. Regions cannot be painted as irregular shapes on any interface.
C. The hands belong to both regions, so the split does not hold.
D. Regional prompting only works with rectangular halves of the frame.

32 Module 4 · Steering with pictures instead of adjectives

You run an image-prompt adapter at high weight and your prompt stops being followed. Why?

A. The adapter overwrites the text embedding before sampling begins.
B. Image features compete with the text for the model's attention.
C. High adapter weights force the sampler into fewer effective steps.
D. The adapter is built for a different base model and is failing quietly.

33 Module 5 · Repair, extend, assemble

An inpainted hand comes back pointing the wrong way. Which control is wrong?

A. Padding, so the model cannot see the arm it belongs to.
B. Denoise, which is too high for a structural repair.
C. Steps, which are too few for a region this complex.
D. Mask blur, which is blending the old geometry straight back in.

34 Module 5 · Repair, extend, assemble

After eight inpainting passes a region has drifted in colour away from the rest. What is the fix?

A. Lower the mask blur so less of the surrounding colour is sampled.
B. Re-run the whole image through img2img at 0.3 to reunify it.
C. Inpaint it once more at a low denoise to match the surroundings.
D. Flatten and correct it with curves in a raster editor.

35 Module 5 · Repair, extend, assemble

A tool consistently modifies pixels outside your mask. What is going on?

A. The image was saved as JPEG, so the mask edge is compressed.
B. The mask blur is set higher than the padding value.
C. It masks the latent rather than the pixels, so edges bleed.
D. The denoise strength is above 1.0 and overflows the region.

36 Module 5 · Repair, extend, assemble

Why do teeth want a lower denoise than you would expect?

A. Teeth occupy too few pixels for a high denoise to converge.
B. High denoise produces a perfect even set that looks like dentures.
C. The mouth mask always includes lip, which a high denoise destroys.
D. Teeth are rendered by the VAE rather than by the denoising network.

37 Module 5 · Repair, extend, assemble

Automatic face detection and repair is run over a twelve-frame set. What is the risk?

A. It can shift the character's identity slightly between frames.
B. It flattens the grain, so the set no longer matches.
C. It writes over the original PNG and loses the generation metadata.
D. It only detects faces that are larger than 256 pixels across.

38 Module 5 · Repair, extend, assemble

What is the thirty-second test for a ghost left behind by an object removal?

A. Run the image through an upscaler and check the same area.
B. Compare the histogram before and after the removal pass.
C. View the image at 25 per cent and look for a change in composition.
D. Duplicate the layer, apply strong curves contrast, and look.

39 Module 5 · Repair, extend, assemble

A dark-haired subject cut from a green background shows a green rim. Why is more feathering not the fix?

A. Feathering only ever works on masks that were drawn with a hard brush.

B. Edge pixels genuinely contain a blend of both; shrink or decontaminate.

C. The rim is a sharpening halo, which feathering only makes worse.

D. Green sits outside sRGB, so the rim cannot be corrected in a raster editor.

40 Module 5 · Repair, extend, assemble

What prompt do you give a tiled diffusion upscale, and why?

A. A prompt describing the single most detailed region of the frame.

B. The original scene prompt, so that every single tile knows what the subject is.

C. A short one naming medium and finish, because tiles see only themselves.

D. An empty prompt, so the model cannot add anything at all.

41 Module 6 · The same thing, twelve times

What does a one-photograph identity adapter actually give you?

A. Nothing usable, since one photograph is below the minimum.

B. The same face only when the pose matches the reference exactly.

C. The person's face, accurate enough to stand up to a passport comparison.

D. A family resemblance: broad geometry, with texture and age drifting.

42 Module 6 · The same thing, twelve times

Why is a character sheet shot in flat, neutral light on a plain background?

A. It separates the identity from everything that will change later.
B. Flat light produces fewer artefacts during LoRA training.
C. It matches the project's anchor image, which is graded last.
D. Neutral light is the only condition identity adapters were trained on.

43 Module 6 · The same thing, twelve times

The profile is the hardest angle to build from a front view. Why?

A. Profiles need a higher adapter weight than the model permits.
B. Adapters were trained mostly on frontal photographs.
C. There is least information about a profile in a front view.
D. Pose controls cannot express a ninety-degree head turn.

44 Module 6 · The same thing, twelve times

You save every epoch during LoRA training. What is that for?

A. So that training can be resumed if the free session disconnects partway.
B. So you can find the middle epoch, where identity and prompt both work.
C. So you can merge the epochs into one stronger adapter.
D. So the learning rate schedule can be tuned between runs.

45 Module 6 · The same thing, twelve times

Five frames from one burst sit in a training set of twenty. What does that do?

A. The set is effectively smaller, and what they share counts five times.
B. It causes the trainer to error on duplicate hashes.
C. It improves the LoRA, since repetition strengthens the concept being learned.
D. It raises the effective resolution of the set.

46 Module 6 · The same thing, twelve times

A LoRA needs a strength of 1.0 before it shows up at all. What does that tell you?

A. It is overtrained, which is why it resists lower strengths.

B. The base checkpoint is a merge and has moved away from it.

C. It was trained at too high a network rank for this particular checkpoint.

D. It is undertrained; you probably picked too early an epoch.

47 Module 6 · The same thing, twelve times

You have one acceptable generated version of a concept product. What do you do next?

A. Use it as an image-prompt reference at weight 1.0 in every frame.

B. Generate it again in each new scene using the same prompt and the same seed.

C. Cut it out and composite it everywhere, as you would a photograph.

D. Train a LoRA on that single image so it can be prompted.

48 Module 6 · The same thing, twelve times

Repair and compositing took 35 per cent of a twelve-image campaign and generating the frames took 6. What does that imply about buying a faster card?

A. It halves the schedule, since generation is the bottleneck.

B. It barely moves the total; the work is before and after generation.

C. It has no effect at all, because repair also runs on the GPU.

D. It matters most for the contact sheets, which dominate the image count.

49 Module 7 · Finishing and handing over

Which class of upscaler is honest and never helps?

A. Bicubic and Lanczos resampling.

B. Real-ESRGAN and SwinIR super-resolution.

C. Ultimate SD Upscale tiled diffusion.

D. GFPGAN and CodeFormer face restoration.

50 Module 7 · Finishing and handing over

Why is upscaling always the last step?

A. Colour profiles are lost if the file is enlarged after conversion.

B. Grain must be added before enlargement or it disappears.

C. Upscalers will only accept flattened files that have no layers in them.

D. It bakes errors in at four times the size and multiplies later work.

51 Module 7 · Finishing and handing over

GFPGAN and CodeFormer restore facial detail well. What is the caution?

A. They only work on faces larger than half the frame.

B. At high fidelity settings they smooth skin into plastic.

C. They change eye colour on any face that is not frontal.

D. They cannot be run without an Nvidia card of 12 GB or more.

52 Module 7 · Finishing and handing over

Why is serving a 6000-pixel hero image to a phone a real cost rather than harmless generosity?

A. It breaks the 2× rule for high-density displays.

B. Browsers are unable to decode images above 4096 pixels on mobile hardware.

C. Page weight is paid by the people least able to afford it.

D. The platform re-encodes it and introduces banding.

53 Module 7 · Finishing and handing over

Why does 100 per cent K alone print as a washed dark grey over a large area?

A. Because the profile converts K to a neutral grey on conversion.

B. Because uncoated stock absorbs black more than it absorbs other inks.

C. Because the black ink is cheaper and is therefore laid down more thinly.

D. One ink cannot cover a large area densely; a rich black uses four.

54 Module 7 · Finishing and handing over

A headline over a busy photograph is failing contrast. Which remedy does the course rank last?

A. A hard outline or drop shadow on the type.

B. A soft black-to-transparent gradient behind the words.

C. A strong blur on the region under the type.

D. Move the type to the quietest area of the frame.

55 Module 7 · Finishing and handing over

An image sits purely as decoration behind a headline. What is the correct alt text?

A. The headline text, repeated for screen reader users.

B. A short description of the image's subject matter.

C. An empty alt attribute, so screen readers skip it.

D. A note that the image is decorative and AI-generated.

56 Module 7 · Finishing and handing over

A client asks for a different expression on the subject. What category is that, and what does it cost?

A. Adjustment; minutes, if the file is layered.

B. Regional; thirty to ninety minutes, and nearby areas may shift.

C. Structural; a new frame and half a day of repeated work.

D. Not possible; expression is fixed by the seed at generation time.

Answers

The key is grouped by module. Each entry gives the page of the question, the question cut short, the right answer and why. For a lesson check it also says why each wrong option tempts. That part is the teaching, not the marking.

1 The kit, and what each piece of it costs you 380

2 From a request to a specification you can generate against 387

3 Getting to a frame worth keeping 394

4 Steering with pictures instead of adjectives 401

5 Repair, extend, assemble 407

6 The same thing, twelve times 413

7 Finishing and handing over 419

Examination 425

Module 1 · The kit, and what each piece of it costs you

The module starts on p. 9.

MODULE 1 TEST

Module 1 test, Q1, p. 64
You download FLUX.1 [dev], run it on your own card, and bill a client for the images. Which statement is accurate?
Answer A: It breaches the model licence, whatever the position on who owns the pictures.
Why: The dev weights carry a non-commercial licence, and running them locally does not change that. Whether you own the output is a separate, unsettled question that differs by country; whether you are licensed to use the model has a definite answer you can read on the model card in two minutes.

Module 1 test, Q2, p. 64
A client approves a bottle shot made on a preset platform, then asks for the light to come from behind the glass instead of the…
Answer D: The lighting decision lives inside the preset and is applied after your words.
Why: A preset is a real workflow somebody froze: a chosen model, a prompt scaffold you never see, probably an adapter, a fixed ratio and a lighting description. Its opinion is applied after your words and overrides them, and there is no point halfway between two presets because they are discrete choices.

Module 1 test, Q3, p. 65
Your first generation after switching checkpoints takes four minutes; every later one on the same model takes fifteen seconds. What is the likeliest cause?
Answer B: System RAM is short, so loading the checkpoint is swapping to disk.
Why: Loading a large checkpoint usually means reading it into system memory first. A machine with 8 GB of RAM swaps to disk and takes minutes to do what a 32 GB machine does in twenty seconds. If the first generation after a model switch is glacial and later ones are fine, it is loading rather than inference.

Module 1 test, Q4, p. 65
Generation reaches one hundred per cent and then fails with CUDA out of memory. Which fix addresses the actual peak?
Answer C: Turn on tiled VAE, because the decode is the peak rather than the sampling.
Why: A failure that arrives after the progress bar completes is the decoder, not the sampler. Tiled VAE splits the decode into overlapping tiles so the final step stops being the memory peak, and it fixes this without touching resolution or batch size.

Module 1 test, Q5, p. 65

A character LoRA that was exact on fp16 is a near-miss on a Q4 model. What is happening, and what should you not do about…

Answer D: Rounded weights land the LoRA's correction slightly off; do not simply raise its strength.

Why: A LoRA encodes a correction to specific weights. When those weights have been rounded to a coarser grid the correction lands slightly off, so the effect is weaker or subtly wrong. Raising the strength over-applies the style while still missing the identity, so test once at a higher precision before spending an hour tuning.

Module 1 test, Q6, p. 66

A client approves an image in March and asks for a small change in September. The same prompt and seed give a different picture. Which…

Answer A: The model hash, the interface commit, and every LoRA with its strength.

Why: Model names are reused constantly and interfaces change defaults between versions. The facts that make an image reproducible are the model hash rather than its name, the interface version as a commit, the sampler and scheduler with steps, CFG and seed, every LoRA with its filename and strength, and the resolution and any upscaler.

THE LESSON CHECKS

Lesson 1, p. 14

A freelancer downloads FLUX.1 [dev], runs it on their own machine with no internet connection, and invoices a client for the resulting images. What is…

Answer C: The [dev] weights are released under a non-commercial licence, so billing a client breaches the model's terms whatever the copyright position on the images turns out to be.

Why: The licence on a model's weights governs whether you may use it commercially, and it is a completely separate question from who owns the picture it made.

A Nothing is wrong. A licence attached…: Reads a model licence as governing distribution only, when it also governs what the outputs may be used for.

B The images are unlikely to attract…: Confuses whether the output is protectable with whether the freelancer was permitted to run the model at all.

D Generating locally strips the Content Credentials…: Answers a provenance question when the question was about permission to use the model.

Lesson 2, p. 17

A team produces 40 product cards a week on a preset platform with no complaints. Then a client asks for one card where the light…

Answer B: The preset has already fixed the lighting decision and applies it regardless of the prompt, so the fix is a tool that exposes lighting directly, or a manual relight in a raster editor.

Why: Recipe-driven tools trade control for speed, and the bill arrives at the first revision, when the thing the client wants changed is the thing the preset already decided.

A The underlying model is too small…: Blames model capacity for a control the tool never exposed in the first place.

C The prompt needs to name the…: Assumes prompt wording can override conditioning that the preset applies after the prompt.

D The preset needs retraining on backlit…: Treats a packaged workflow as a model the user can retrain, rather than a fixed set of settings.

Lesson 3, p. 22

Somebody generates about 150 images a month and keeps hitting a hosted service's refusal filter on ordinary historical war imagery for a school project. Which…

Answer A: Control, not cost: at this volume hosting is a few dollars a month, so the real question is whose policy decides what they may make.

Why: The choice between a hosted service, your own card and a free cloud notebook is decided by how many images you make per month and how much control you need, and the crossover is usually somewhere around a few thousand images.

B Cost, because 150 images a month…: Treats the decision as volume-driven when 150 images at a few cents each is a few dollars a month; a card would take many years to repay at that rate.

C They should move to a free…: Half right about the filter but ignores persistence: nothing survives the session, so every run re-downloads the checkpoint and the setup, which is a poor fit for steady low-volume work.

D They should rewrite the prompts, since…: Assumes filters are precise. They combine a keyword layer with an output classifier and are deliberately over-inclusive, so ordinary historical language trips them regardless of intent.

Lesson 4, p. 27

An SDXL image has correct composition and correct framing but comes out washed out with a purple cast on every generation, regardless of prompt or…

Answer B: The VAE is broken or missing, so a correctly denoised latent is being decoded wrongly.

Why: A checkpoint, a VAE, a LoRA and an embedding are different file types with different sizes and different folders, and almost every "it loaded but the image is grey mush" problem is one of them in the wrong place.

A A LoRA trained against a different…: A base mismatch usually errors outright or produces incoherent content, not a consistent colour cast with intact composition.

C The guidance scale has been set…: High guidance does damage colour, but toward burnt contrast and clipping, and it would vary with the prompt rather than appearing identically on every generation.

D The checkpoint was downloaded as a…: Confuses a container format with numeric precision; the two extensions store identical weights, and the difference is only whether loading can execute code.

Lesson 5, p. 31

A studio produces the same eleven-step retouch-and-composite process on roughly forty product images a month, handed between two people. Which interface choice best fits, and…

Answer B: ComfyUI, because the eleven steps become one saved graph both people run identically.

Why: The interface you learn decides which problems are easy, and the right choice is the simplest one that can express the workflow you actually need — which for finished client work usually means a node graph eventually.

A Fooocus, because its curated defaults produce…: Consistency here comes from repeating a specific process, not from defaults; Fooocus deliberately hides the steps the studio needs to control and share.

C AUTOMATIC1111 or Forge, because a settings…: Fine for one operator on one day; the process lives in memory and notes, which is exactly what fails when work is handed between two people over months.

D InvokeAI, because its unified canvas is…: True about single-image repair, and the wrong axis: the problem stated is repeatability across forty images and two operators, not canvas ergonomics.

Lesson 6, p. 35

A local SDXL generation reaches 100% on the progress bar and then fails with a CUDA out-of-memory error. What does the timing tell you?

Answer C: The peak is the VAE decode that runs after sampling, so tiled decoding is the targeted fix.

Why: Almost every failed local install is one of four specific errors, each with a one-line cause, and knowing them turns a lost weekend into twenty minutes.

A Nothing useful, because an out-of-memory failure…: Discards the most diagnostic piece of information available, and resolution is only one of four levers, applied here to the wrong stage.

B The batch size is too high…: Batching multiplies memory throughout sampling, not at the end, so a batch problem fails early rather than after the bar completes.

D The checkpoint itself is too large…: Loading happens before sampling, so a checkpoint that did not fit would have failed at 0%, not at 100%.

Lesson 7, p. 40

A frame looks right at 25 steps with Euler a. Rendering the same seed and prompt at 45 steps for the final produces a noticeably…

Answer B: Euler a is ancestral: it adds fresh noise at each step, so it never converges and a different step count follows a different path.

Why: Doubling the step count past roughly 30 buys almost nothing on a standard model, while a distilled model does a usable image in 4 to 8 steps, so measure iterations per second and spend your time where the curve is still steep.

A The extra steps revealed fine detail…: Describes refinement, which would not move composition; added steps on a converging sampler change texture rather than what the picture is of.

C The seed only sets the initial…: Overstates it: with a convergent sampler the same seed and prompt at 25 and 45 steps give recognisably the same picture, which is exactly why the ancestral distinction matters.

D The scheduler re-spaces the noise timetable…: The schedule is indeed re-spaced, but convergent samplers still land in almost the same place; blaming the schedule alone predicts instability everywhere, which is not what happens.

Lesson 8, p. 45

Someone running a Q4 quantised model finds their character LoRA gives only a loose resemblance. They raise LoRA strength from 0.8 to 1.2 and the…

Answer C: Quantisation rounded the base weights the correction is computed against, so it lands off-target and strength amplifies the error too.

Why: Quantisation and offloading trade quality or speed for memory in known amounts — fp8 costs almost nothing, Q4 costs a little detail, and CPU offload costs seconds rather than quality.

A The LoRA was trained at too…: A plausible separate defect, but it would misbehave identically at full precision, which is the test this situation calls for.

B Strength above 1.0 is outside the…: Values above 1.0 are unusual but legal and often useful; the damage here is over-application of what the LoRA does carry, not an invalid range.

D The quantised build uses a different…: Key mismatch does occur across base families and produces a warning or a no-op, not a partial resemblance that scales with strength.

Lesson 9, p. 50

A charity campaign about domestic violence needs an image showing bruising. Two hosted services refuse it. What is the professionally sound response?

Answer A: Rephrase clinically, and if that fails, plan the sensitive element as a local render or a photograph to composite.

Why: A refusal is produced by a keyword layer, an output classifier and an account-level policy stacked together, all tuned for the platform′s liability rather than your brief, which is why the same prompt passes on one service and fails on another.

B Split the description into fragments that…: This is filter evasion by another name; it breaches the terms you agreed to and risks losing the account mid-campaign, with no argument you can make to the client afterwards.

C Accept that this imagery is simply…: Gives up too early: the prompt filter is one of three layers and is over-inclusive by design, so clinical phrasing or a composited element usually resolves it without altering the brief.

D Move to whichever service has the…: Shopping for permissiveness treats a liability decision as a feature comparison, and leaves the project dependent on a policy that can tighten overnight.

Lesson 10, p. 54

An approved image cannot be reproduced six months later despite identical prompt, seed, sampler, steps and CFG. Which recorded fact would most likely have explained…

Answer B: The model hash, because checkpoints are often re-uploaded under the same filename with different weights.

Why: Diffusion output depends on the exact version of the interface, the model hash and the sampler implementation, so recording those four or five facts with each approved image is what makes a client revision six months later a ten-minute job.

A The random number generator differs between…: A genuine source of variation, but it is a setting rather than a drift, and it would not change between two runs on the same unchanged machine.

C The original output resolution, because generating…: Resolution does change the image completely, but it is part of the parameters already stated as identical in the question.

D The negative prompt, which a number…: Most interfaces do save it, and like the other parameters it is text the operator still holds; the thing that silently changed is the file the name points to.

Lesson 11, p. 57

A freelancer exhausts a month's credit allowance in two days on a single hero image and still does not have a usable frame. What is…

Answer B: The composition should have been chosen from batches of cheap, small, low-step renders before anything was rendered at final quality, and after roughly six full-quality attempts the remaining distance is editing work rather than generation.

Why: Choose the frame on cheap low-step renders and spend full quality only on the winner; after about six full-quality attempts the remaining distance is editing work, not generation.

A The plan is too small for…: Buys more of a wasteful process instead of changing the process.

C They should move to running an…: Shifts the cost onto hardware without reducing the number of wasted full-quality renders.

D A higher step count would have…: Assumes quality keeps scaling with steps well past the point where the sampler has converged.

Module 2 · From a request to a specification you can generate against

The module starts on p. 67.

MODULE 2 TEST

Module 2 test, Q1, p. 114

Why does an SDXL render at 1024 × 2048 often contain two torsos?

Answer C: The canvas is far outside the trained buckets, so two regions each resolve as a centre.

Why: Training images are grouped into buckets at a roughly constant pixel count, and inside those the model has seen thousands of examples of how a subject fits the frame. Stretch the canvas well beyond them and two regions independently decide they are each looking at the centre of a portrait.

Module 2 test, Q2, p. 114

You need 1200 × 630 and the nearest SDXL bucket is 1344 × 768. What is the recommended route?

Answer B: Generate at the bucket and outpaint the sides rather than cropping top and bottom.

Why: Cropping a 1.75 frame down to 1.9 throws away composition you wanted, so the extra width is built rather than taken. Generating outside the bucket produces duplicated subjects, generating larger than the trained pixel count produces two horizons, and stretching is visible.

Module 2 test, Q3, p. 114

What does designing a frame for crop safety actually cost you?

Answer A: A more centred, more conservative frame than one composed for a single format.

Why: A frame built to survive five crops is by definition more centred than one composed for one place. That is a real trade rather than a free technique, so it belongs on images that will travel — social assets, stock, anything a marketing team re-uses — and not on a hero that appears in exactly one slot.

Module 2 test, Q4, p. 115

Why does "no logos" in the negative prompt not protect a client from a trademarked shape appearing on background signage?

Answer D: The negative prompt biases the outcome; it does not forbid anything.

Why: The negative prompt produces a second prediction the sampler steers away from, in proportion to the guidance scale. It reduces how often a thing appears. For something that would cost a client a legal letter, a probabilistic reduction is a hope with a technical-sounding name, so the guarantee is a written list a person checks on the finished file.

Module 2 test, Q5, p. 115

A brand red is specified as #E4002B. What is the reliable way to hit it?

Answer C: Generate the element neutral and set the exact value in an editor afterwards.

Why: The prompt goes through a text encoder that produces a representation of meaning, and a hex string is tokenised into fragments that carry none. There is no path in the architecture by which a specific RGB triple can be requested, so the control is after generation, in pixel space, where a colour is a number you can set.

Module 2 test, Q6, p. 115

A client approves one hero image and then asks for five more like it. Why is that a different job rather than a small extension?

Answer B: The visual system has to be built backwards from a frame composed without one.

Why: A set is a consistency problem rather than a search. "Five more like it" means deriving the light, lens, palette, crop convention and finish retrospectively from an image that was never made to embody them, which is harder than fixing them before the first render.

THE LESSON CHECKS

Lesson 12, p. 72

A client asks for a launch image for "socials and the website". What is the most useful first action?

Answer A: Ask for the exact placements and their pixel sizes, since ratio, detail and type placement all follow from them.

Why: The final placement fixes the dimensions, the crop, the amount of detail that will ever be visible and the file format, and every one of those decisions is cheaper to make before generating than after.

B Generate a square image, because a...: A square is a compromise that satisfies nothing well; it crops badly to both 9:16 and 1.91:1, and the hedge is chosen instead of the information that would remove the need for one.

C Generate at the highest resolution the...: Resolution is not the constraint; aspect ratio and crop-safe composition are, and no amount of extra pixels makes a 4:5 frame work as a 1.91:1 card.

D Begin with the visual concept and...: Sounds like sensible creative sequencing, but approving a composition before knowing its frame means the approved thing may be impossible to deliver in the required shape.

Lesson 13, p. 76

An SDXL render at 832 × 1600 produces a figure with two sets of shoulders. What is the correct response?

Answer D: Generate at a native bucket such as 832 × 1216, then outpaint upward in steps to the final height.

Why: Models are trained at a specific set of resolutions, and generating far outside those buckets produces duplicated subjects and stretched anatomy, so you pick the nearest trained ratio and crop to the delivery size afterwards.

A Raise the guidance scale so the…: Guidance changes how hard the model follows the text, not how it handles a canvas shape it has no training examples for; the duplication persists at every CFG value.

B Add "one person, single subject, solo"…: Occasionally helps by luck, but it is fighting a geometric cause with vocabulary; the second torso comes from canvas size rather than from an ambiguous description of the subject.

C Generate at 832 × 1600 with…: Extra steps refine what is already being formed, so they will produce a better-finished pair of shoulders rather than removing the second one.

Lesson 14, p. 80

A 4:5 social image keeps losing the subject's face when platforms crop it to a square thumbnail. What is the underlying composition error?

Answer B: Important content was placed in the band that only survives the original ratio, instead of the central core.

Why: Plan the frame as concentric zones — a core that survives every crop, a middle that survives most, and an outer band that exists to be thrown away — because the image will be re-cropped by people and platforms who never see your original.

A The image was generated at a…: Bucket mismatch produces duplicated or distorted anatomy rather than a subject positioned too near an edge; the framing here is a compositional choice, not an artefact.

C The resolution is too low, so…: Invents a mechanism: platform crops are usually fixed or face-detected, and neither behaves differently because an image has more pixels in it.

D The delivered file should have been…: Solves it by giving up the 4:5 frame entirely; the point of safe areas is to serve several crops from one strong composition rather than defaulting to the weakest one.

Lesson 15, p. 84

What makes a labelled six-image reference pack more useful than a twenty-image mood board?

Answer B: Each image names the one attribute it contributes, making it checkable at sign-off and routable to a conditioning method.

Why: A mood board of vibes cannot be acted on, but six references each labelled with the single thing it is there to demonstrate can be used directly as conditioning and as the basis for sign-off.

A Six images can be supplied to…: Invents a technical ceiling; the gain comes from the labelling, not the count, and you rarely condition on all six simultaneously anyway.

C A smaller set reduces the risk…: Fewer references do not reduce resemblance risk; heavy conditioning on one image raises it regardless of how many others sit beside it in the folder.

D Twenty references take substantially longer to…: Treats a decision-making tool as a scheduling convenience, and misses that an unlabelled pack of six is no more actionable than an unlabelled pack of twenty.

Lesson 16, p. 87

Why should "no visible brand logos" be a delivery checklist item rather than a negative prompt entry?

Answer A: Because a negative prompt shifts the odds rather than enforcing a rule, and the consequence of one logo slipping through is legal rather than aesthetic.

Why: A negative prompt is a weak statistical nudge rather than a guarantee, so anything that would be genuinely damaging in the final image belongs on a written checklist that a person verifies before delivery.

B Because negative prompts stop working reliably…: Invents a threshold; negative guidance weakens at low CFG but the real point is that it is a bias at every setting, not a prohibition at some of them.

C Because logos are usually generated in…: The negative applies to the whole prediction rather than to a favoured region, so this describes a spatial mechanism that does not exist.

D Because naming a brand in the…: Confuses describing something internally with using a mark in commerce, and would leave the actual failure — a logo appearing in the delivered file — entirely unaddressed.

Lesson 17, p. 90

Why can a diffusion model not reliably produce an exact hex colour on request?

Answer A: Because the text encoder turns the prompt into a semantic embedding, so there is no channel by which a specific numeric value reaches the pixels.

Why: Diffusion models cannot be instructed to produce an exact colour, so brand colour is achieved by generating in a neutral or near palette and then grading or compositing the exact value in afterwards.

B Because the VAE decode from latent…: The decoder is not colour-accurate in a strict sense, but it is not the barrier; the value never existed as a target in the first place, so there is nothing for a transform to shift.

C Because the model's training captions almost…: Treats a representational limit as a data gap; even a perfectly captioned dataset leaves colour as a distribution over lighting and material rather than a settable coordinate.

D Because guidance scale controls saturation as…: Guidance does affect saturation, which makes this feel right, but the mismatch appears at every guidance value including ones where saturation is well behaved.

Lesson 18, p. 95

Why does a twelve-image set need a different method from a single hero image?

Answer C: Because the frames are judged against each other, so the work shifts from searching for one good frame to enforcing constants across all of them.

Why: A single image is a search problem and a set is a consistency problem, and they need different budgets, different methods and a decision made before any generating starts.

A Because generating twelve images costs roughly…: Compute is rarely the binding constraint at this scale, and the harder problem is that constrained frames need more attempts each, not that renders are expensive.

B Because each image in a set…: Viewing time varies by placement rather than by set membership, and a visibly weaker frame damages a set more than it would damage a standalone image.

D Because a set is more likely…: Crop safety is a real constraint but an independent one; a single hero reused across placements faces it just as much as a set does.

Lesson 19, p. 99

What distinguishes the frames that should be photographed from the ones that should be generated?

Answer B: Whether the image asserts something about a specific real thing that a viewer could check against reality.

Why: Anything that must be exactly itself — a real product, real premises, real staff, food you are selling — is faster, cheaper and safer to photograph than to generate, and recognising those frames early is a professional skill rather than a failure.

A Whether the subject contains fine detail…: Names a real weakness but the wrong criterion: a generated abstract background with no text at all is still fine, while a plain unbranded product still needs to be the actual one.

C Whether photographing it would be cheaper…: Cost is a secondary consideration; a cheap generated staff photo is still a fabrication, and an expensive photographed environment may still be the wrong call.

D Whether the final image will be…: Display size affects how much repair a frame needs, not whether the image is making a factual claim that has to be true.

Lesson 20, p. 104

What does an image spec actually accomplish at review time?

Answer C: It turns a disagreement about taste into a comparison against a record both sides agreed to beforehand.

Why: A signed one-page spec that names dimensions, ratio, forbidden content, the reference pack and what will be photographed rather than generated converts later disputes from matters of taste into matters of record.

A It transfers responsibility for any content…: Misreads a shared working record as a liability shield; the forbidden list still has to be checked by whoever produces the file.

B It documents the exact technical parameters…: Reproducibility comes from recorded model hashes and saved files, not from a brief; a spec describes intent rather than settings.

D It allows the number of revision…: The rounds clause is genuinely useful, but it is one line of a document whose broader function is to make every later disagreement checkable.

Lesson 21, p. 108

An agency in India delivers a generated photograph of a smiling "customer" for a client's testimonial page, with no label, arguing that the tool's terms…

Answer C: Ownership and disclosure are separate questions; a synthetic image presented as a real customer is precisely what labelling rules address, India's IT Rules prescribe a visible label, and the client should have been told in writing that the asset is generated.

Why: Being licensed to use a model, owning its output, and having to disclose that output is generated are three separate questions, and only the ownership one is genuinely unsettled.

A Nothing, provided the tool's terms genuinely…: Treats a contract term about ownership as an answer to a duty that arises from disclosure law instead.

B The file only needs a C2PA…: Relies on metadata that a screenshot or a re-upload removes, where the rule prescribes something a viewer can actually see.

D Synthetic-media rules cover video deepfakes of…: Assumes these rules are limited to video and to impersonation of named people, rather than to synthetic content presented as real.

Module 3 · Getting to a frame worth keeping

The module starts on p. 117.

MODULE 3 TEST

Module 3 test, Q1, p. 168
A ninety-word prompt on an SDXL checkpoint puts all the right things in none of the right relationships. What is the mechanism?

Answer A: The prompt is encoded in 77-token chunks with no attention across the boundary.

Why: Interfaces get past CLIP's 77-token window by splitting a long prompt into chunks, encoding each separately and concatenating the results. There is no attention across a chunk boundary, so a relationship described in words eighty to ninety has no path by which it can bind to a subject named in word five.

Module 3 test, Q2, p. 168
Somebody tells you that adding "8k, masterpiece, highly detailed" improves a modern checkpoint. How do you settle it, and what is the likely finding?

Answer D: Fix the seed, generate with and without, and expect almost no difference.

Why: Resolution is set by the image size rather than by words, and those tags were tuned for particular SD 1.5 fine-tunes. A fixed seed with exactly one variable moved is the only comparison that attributes a difference to the change, and on most current checkpoints there is almost none to attribute.

Module 3 test, Q3, p. 169
Why does leaving light unspecified produce the recognisably "generated" look?

Answer B: Unspecified light returns the model's average, which has no dominant source.

Why: Left unspecified, the model gives you the average of millions of photographs, and averaging light produces soft, directionless illumination with weak shadows. Plastic is a light problem rather than a texture one: give the scene one strong source and a stated direction and most of it goes.

Module 3 test, Q4, p. 169
Why is "risograph, two colours, slight misregistration" more stable across seeds than naming a living illustrator?

Answer C: Every risograph print in the training data shares the same physical signature.

Why: Process vocabulary describes how the image was made, which is exactly what produced the consistent appearance in the training data, so the phrase has a tight and steady meaning. A name has a diffuse one — early work, late work, the most-reproduced pieces, imitators, whatever the captions happened to say.

Module 3 test, Q5, p. 169

Which fault should not influence which frame you take forward from a contact sheet?

Answer D: Hands with the wrong number of fingers.

Why: Hands are always repairable: mask, regenerate at full resolution, or composite a photographed one. Composition, light direction and structural impossibility are not, because fixing them means generating again. The most expensive mistake here is choosing a frame with beautiful hands and mediocre composition.

Module 3 test, Q6, p. 170

Somebody posts a seed and a prompt, you match both, and your render is a different picture. What is the usual explanation?

Answer A: Something in the configuration differs — sampler, resolution, or CPU against GPU noise.

Why: A seed is only reproducible inside one configuration. The model, the sampler, the scheduler, the step count on ancestral samplers, the resolution, the interface version and whether noise is generated on CPU or GPU all break it. A seed is a bookmark in your own setup, not a coordinate in a shared space.

THE LESSON CHECKS

Lesson 22, p. 123

A ninety-word prompt describes a woman in a blue coat and a man in a red scarf, and the model keeps swapping the colours. Adding…

Answer B: Attribute binding weakens across the 77-token chunk boundary, so a long prompt loses the link between a subject and its attributes.

Why: Writing a prompt as seven named slots makes it debuggable, because when the image is wrong you can change one slot and know which part of the description caused the change.

A Colour words are weakly represented in…: Colour binds tightly enough in short prompts, which is the clue that length rather than the colour vocabulary is what broke it.

C The guidance scale is too low…: Guidance strengthens adherence to the prompt as a whole; raising it on a prompt with broken binding produces a more emphatic version of the same swap.

D The two subjects are too similar…: A real effect for genuinely similar subjects, but it would not depend on prompt length, and the same pair described briefly usually binds correctly.

Lesson 23, p. 127

Why do focal-length terms like "35mm" and "85mm" change an image more reliably than "8k, masterpiece"?

Answer C: Because focal lengths appeared in enormous numbers of training captions attached to a consistent look, whereas quality tags did not.

Why: Focal length, aperture, camera height and shot size measurably change composition because they appeared in the captions of millions of training photographs, while quality words like "8k" and "masterpiece" mostly do nothing on modern models.

A Because numeric terms are tokenised more…: Tokenisation is not the mechanism; "35mm" works because of what it was attached to in captions, not because digits encode more crisply than letters.

B Because focal length physically determines perspective…: The model has no geometry of lenses; it reproduces the appearance that accompanied those captions, which is why the resulting blur sometimes lands at the wrong depth.

D Because quality tags are usually placed…: Position has some effect on CLIP-based models, but moving "masterpiece" to the front does not make it work, which is the test that rules this out.

Lesson 24, p. 131

Generated portraits look plastic despite a detailed subject description. What is the most likely cause?

Answer B: No light is specified, so the model produces its soft directionless average, which reads as plastic.

Why: Light is the single highest-leverage slot in a prompt, and leaving it unspecified is the main reason generated images look generically generated.

A The checkpoint is a community merge…: Merges do drift toward a glossy house look and this is worth checking, but the same checkpoint produces convincing skin the moment a directional source is named.

C The prompt lacks skin-texture vocabulary such…: Texture words help a little at the pixel level, but a face with real pores under flat directionless light still reads as plastic, because the problem is form rather than surface.

D The guidance scale is set too…: High guidance does flatten and over-contrast an image, but it produces a burnt look rather than the specific weightless quality being described.

Lesson 25, p. 135

Why does "risograph print, two colours, slight misregistration" reproduce a look more consistently than naming a contemporary illustrator?

Answer C: Because every risograph print in the training data shares the same physical characteristics, while an artist's name covers decades of varied work and imitations.

Why: A look is more reliably reproduced by naming its process, era, materials and printing than by naming an artist, and the process description also avoids the ethical and commercial problem of trading on a living person's name.

A Because process terms are shorter, so…: Length is incidental; a two-word artist name is shorter still and remains far less stable across seeds.

B Because artist names are increasingly filtered…: Some services do filter names, but the instability predates any filtering and appears on unfiltered local weights as well.

D Because process terms describe the image…: Invents a staged mechanism; denoising does not separate surface from content into distinct phases that can be targeted independently.

Lesson 26, p. 140

Why judge exploration frames as thumbnails on a grid rather than at full size one at a time?

Answer B: Because small size hides repairable detail and forces judgement onto composition, and relative comparison is more reliable than absolute.

Why: Judging a grid of many low-cost renders at once is faster and more accurate than judging full-quality images one at a time, because composition is visible at thumbnail size and commitment bias is not.

A Because thumbnails load faster, which allows…: Review speed is a minor benefit; the reason the method works is what small size makes invisible, not how quickly the files open.

C Because low-step renders are too unfinished…: Sensible-sounding but backwards: the artefacts are ignored because they are irrelevant to the decision, not because the images are unfinished.

D Because a grid layout makes it…: Structural faults are one of the three questions, not the purpose; a single full-size frame shows a duplicated torso perfectly well.

Lesson 27, p. 145

Someone finds a good composition at seed 418, then switches from 832 × 1216 to 1024 × 1024 at the same seed and gets an…

Answer D: The seed builds a noise field shaped to the output size, so a different resolution is a different starting field.

Why: A seed fixes the starting noise, so changing it and the prompt together means you cannot attribute the difference to either, and seed-locked comparison is what turns guessing into testing.

A The aspect ratio changed, so the…: Bucket choice affects how a composition is arranged, but it would not replace the image entirely; even the subject and palette are different here.

B Seeds are only valid within a…: No such session scoping exists; a seed reproduces reliably across restarts as long as the configuration matches.

C Larger images consume more random numbers…: Closer to the mechanism but wrong about the consequence: the field is a different shape entirely rather than the same field extended with extra values.

Lesson 28, p. 149

Why is a fixed seed necessary when testing whether a prompt change helped?

Answer C: Because a different seed produces a different image regardless of the prompt, so any difference cannot be attributed to the change.

Why: A comparison is only informative when one variable moves, and the habit of changing prompt, sampler and steps together is why so much accumulated knowledge about image generation turns out to be wrong.

A Because random seeds produce a wider…: Describes a variance problem and implies more samples would fix it; the deeper issue is that the change and the noise are confounded in a single pair.

B Because interfaces cache the previous generation…: Caching is not what happens, and speed has nothing to do with whether the comparison is valid.

D Because some seeds are inherently better…: There is no compatibility between seeds and prompts; a seed is just a starting noise field with no affinity for particular descriptions.

Lesson 29, p. 153

Three attempts at "seven birds on a wire" give five, nine and four birds. What does the consistency of the failure tell you?

Answer C: That quantity is represented as a distribution rather than counted, so no rephrasing fixes it and the number has to come from compositing.

Why: Some requests fail for structural reasons rather than for want of a better prompt, and the professional skill is recognising them inside three attempts and switching method instead of continuing.

A That the prompt is ambiguous about…: Adds detail to a description that was already clear; the count is unreliable because nothing in the model enumerates objects, not because the arrangement was underspecified.

B That the seed is unlucky, and…: Occasionally true by chance, and it is the reasoning that produces the sunk-cost hour; hunting a distribution for an exact value is not a method.

D That the guidance scale is too…: Guidance amplifies the whole conditioning signal, so raising it produces a more confident wrong number rather than a correct one.

Lesson 30, p. 157

Why does feeding an interrogator's caption back into a generator not reproduce the source image?

Answer D: Because a caption describes appearance rather than inverting the generation, so arrangement, proportion and exact light are never captured.

Why: An interrogator recovers usable vocabulary from a reference image but never recovers the image, because it is describing appearance rather than inverting the generation process.

A Because interrogators are trained on generated…: They are trained on image-text pairs from the web, largely photographic; the loss is inherent to description rather than a domain mismatch.

B Because the caption omits the seed…: True for a generated source and irrelevant for a photograph, which never had a seed; the caption would still not reproduce it.

C Because interrogators return tags rather than…: Sentence-style interrogators exist and fail in the same way, so grammar is not what is missing.

Lesson 31, p. 162

Two candidate frames: one has excellent composition and badly mangled hands, the other has perfect hands and a flat, centred composition. Which goes forward?

Answer C: The first, because composition cannot be created in repair while hands reliably can.

Why: A frame is worth repairing when its composition, light and structure are right, because those are the things repair cannot create, while hands, small objects and texture always can be fixed.

A The second, since correcting composition through…: Cropping and outpainting adjust the edges of a frame; neither redistributes weight within a composition or adds the directional light that was never there.

B Neither, because a frame requiring substantial…: Over-strict: almost every usable frame arrives with a repair list, and thirty minutes of masked repair is the normal cost of a good composition.

D The first, but only if the…: Adds a condition that is not needed; masked repair at full resolution fixes hands routinely, and compositing is one option rather than a prerequisite.

Module 4 · Steering with pictures instead of adjectives

The module starts on p. 171.

MODULE 4 TEST

Module 4 test, Q1, p. 216
You run img2img at 0.85 and almost nothing of your photograph survives. What should you conclude?
Answer C: The strength is far above the useful range; bisect between 0.3 and 0.7.
Why: Denoise strength is roughly the proportion of the sampling schedule that runs, so at 0.85 the input contributes little beyond a colour distribution. The working range is about 0.4 to 0.6, and you find it by running 0.3, 0.5 and 0.7 on one seed and splitting the interval that looked closest.

Module 4 test, Q2, p. 216
A conditioned image comes back flat and traced. Which setting is most likely wrong?
Answer B: The control runs to the final step instead of ending around 60 per cent.
Why: Composition is decided early in the schedule and fine texture late. Enforcing structure right through the phase where texture forms produces a colouring-in exercise; releasing the control around 60 to 70 per cent gives you the layout you chose and a surface the model built for itself.

Module 4 test, Q3, p. 217
Why does a two-minute painting of flat colour blocks control layout and palette at the same time?
Answer A: Starting from an image skips the early schedule, where layout and colour are set.
Why: Diffusion resolves an image coarse to fine, and starting from an existing image skips the early part of the schedule in proportion to the denoise strength. At 0.6 roughly the first forty per cent never runs, so the broad layout and colour decisions have already been made by your painting.

Module 4 test, Q4, p. 217
You need a different person in exactly this photograph's pose, with everything else free to change. Which control?
Answer D: OpenPose, which keeps joints and discards shape and clothing.
Why: The question is always what must survive and what must be free. OpenPose extracts joints and limbs and discards body shape, clothing, age and identity, which is exactly the split the brief describes. Canny would carry the original person's outline through and depth would carry their volume.

Module 4 test, Q5, p. 217
Four seeds at your current settings come back almost identical. What does that tell you?
Answer C: You have over-constrained; seed variation is the freedom you removed.
Why: Every constraint removes degrees of freedom from the generative process. Four near-identical seeds are the clearest signal that the total is too high, and the remedy is to drop the combined strength by about 0.3 rather than to reach for a different control type.

Module 4 test, Q6, p. 218
One area of a conditioned image looks wrong on every seed while everything else is fine. What is the diagnosis and the fix?
Answer B: Two controls disagree there; make the inputs agree before generating.
Why: A pose skeleton putting an arm where the depth map has a solid wall gives the model two contradictory demands, and you get an arm through the wall, a melted region, or one control silently losing. Overlay the two control images and look at that area. Tuning strengths only decides which constraint wins.

THE LESSON CHECKS

Lesson 32, p. 175

Someone feeds a phone photo of their shop into an image-to-image tool at 0.85 strength and gets a picture with nothing of the shop in…

Answer A: Strength sets how much noise is added before denoising begins, so it controls how much of the original survives; both values sit at the extremes, and the range that keeps the layout while changing the rendering is around the middle, found by bisecting.

Why: Denoise strength decides how much of your reference survives; the useful range is roughly 0.4 to 0.6, and everything above 0.75 throws the reference away.

B The reference is too low-resolution for…: Blames the input's quality for behaviour fully explained by a parameter set to its extremes.

C Image-to-image transfers style but never composition…: Rules out a technique that works, on the evidence of settings that switched it off.

D The prompt is competing with the…: Attributes to prompt conflict what the strength value on its own accounts for.

Lesson 33, p. 179

A designer conditions a render on a scanned pencil sketch using Canny edges at weight 1.0 for the full run. The result looks stiff and…

Answer B: Canny keeps every edge it detects, including paper grain and scan artefacts, and holding the control at full weight for all the steps prevents the model from resolving the drawing into a rendered image; raising the thresholds and ending the control around 60% fixes both.

Why: Conditioning makes composition an input rather than an outcome, and letting the control end around 60% of the steps is what stops the result looking traced.

A Canny is the wrong preprocessor for…: Swaps the tool when the settings, not the choice of tool, produced both problems.

C The sketch was scanned at too…: Blames the input's size for a problem caused by edge thresholds and control scheduling.

D The run needs more denoising steps…: Expects extra compute to overcome a constraint that is being re-applied at every one of those steps.

Lesson 34, p. 183

A designer runs img2img at increasing denoise values trying to shift the subject from the centre to the left of the frame. By 0.8 the…

Answer A: Img2img denoises the whole latent toward the prompt rather than manipulating objects, so position only changes when the source stops surviving.

Why: Running several img2img passes at decreasing denoise strength converges on a finished image more reliably than one pass at a middling value, because each pass only has to make a small correction.

B The denoise value applies uniformly across…: A masked pass would regenerate one region but cannot relocate the subject into it; you would end up with two subjects or an empty space.

C The seed was not held constant…: Fixing the seed changes the starting noise but not the fact that img2img has no mechanism for moving content within the frame.

D The model's attention layers bind the…: Invents a centre-bias mechanism; the subject stays put because the source pixels stay put, not because of any positional preference in the architecture.

Lesson 35, p. 187

Why should a sketch control be set to end around 60–70% of the steps rather than running to completion?

Answer B: Because the final steps form fine texture, and enforcing the drawn structure through them makes the result look traced.

Why: Five minutes of rough drawing photographed on a phone gives the model a composition it will follow, and it requires no drawing skill because the conditioning reads structure rather than quality of line.

A Because control models lose accuracy in…: Backwards: the control remains accurate throughout, and the problem is that it stays accurate during the phase where texture should be forming freely.

C Because running a control to the…: Control cost is roughly constant across steps, and memory has nothing to do with the flat appearance the setting exists to prevent.

D Because ending the control early allows…: The prompt does gain influence, but it will not restructure the composition at that late stage; what it gains is freedom over surface, not over arrangement.

Lesson 36, p. 190

Why does a flat colour-block painting run at denoise 0.6 control the layout of the output?

Answer B: Because the early part of the schedule is skipped, so broad layout and colour are already fixed by the painting when denoising begins.

Why: A crude painting of flat colour shapes run through img2img at low denoise places objects and palette exactly, because the colour distribution of the input survives the parts of the schedule that decide layout.

A Because flat regions of uniform colour…: Describes a property of the input rather than a mechanism of control; a flat region could just as easily be overwritten, and at 0.85 it is.

C Because the model treats the input…: Confuses img2img with a control adapter; there is no separate conditioning channel here, only a partially noised starting image.

D Because colour information survives the encode…: Both survive together at this strength, and the question is about layout persisting, which this answer explicitly denies.

Lesson 37, p. 194

What does depth conditioning from a Blender blockout provide that a hand-drawn sketch cannot?

Answer A: Geometrically correct perspective and true relative scale, because the camera projection is computed rather than estimated by hand.

Why: Rendering a depth map from a few primitives in Blender gives the model correct perspective and object placement that no sketch or prompt can supply, and it is the standard professional route for interiors, architecture and product staging.

B A stronger conditioning signal, since depth…: Depth does carry more information, but scribble conditioning is not weak; the decisive difference is that the geometry is correct, not that the signal is denser.

C Control over materials and surface finish…: A depth pass contains only distance; material comes entirely from the prompt, which is why vague prompts give inconsistent substance across seeds.

D Separation between objects that sit at…: Exactly backwards: depth merges same-distance objects into one grey region, and line art is what keeps their edges apart.

Lesson 38, p. 198

You need to put a different person into the exact pose of a reference photograph, with different clothing, build and setting. Which control type fits?

Answer D: OpenPose, because it extracts joint positions only and discards build, clothing and identity.

Why: Each control type preserves one property of the input and discards the rest, so the choice is made by naming what must survive from the source and what must be free to change.

A Canny, because the edge map captures…: Canny preserves the person's actual silhouette, clothing edges and build, so the new figure would inherit the body you were trying to replace.

B Depth, because it preserves the three-dimensional…: Depth carries body volume as well as limb position, so build and clothing bulk survive alongside the pose.

C Softedge, because its thick soft boundaries…: Closer, and still carries a good deal of the original silhouette; softedge is for "roughly this arrangement" rather than for stripping a pose out of a body.

Lesson 39, p. 202

Why does regional prompting solve attribute swapping between two subjects when prompt emphasis does not?

Answer C: Because the attributes are described in separate conditioning applied to separate areas, so there is no binding to get wrong.

Why: Regional prompting assigns separate text to separate areas of the image, which solves attribute binding failures that no amount of rephrasing a single prompt can fix.

A Because each region is sampled independently…: Overstates the isolation: regions share one sampling process and a base prompt, and it is the separate conditioning rather than separate sampling that does the work.

B Because regional masks are applied in…: Regions act during sampling in latent space, not as a post-process; a pixel-space composite would show hard seams and could not affect what was generated.

D Because emphasis syntax multiplies token weights…: Accurately describes why emphasis fails but not why regions succeed, and the question asks for the second.

Lesson 40, p. 205

A designer needs one image in the visual style of a supplied reference, today. What makes an image-prompt adapter the right choice over training a...

Answer B: An adapter requires no dataset and no training time, which suits a one-off where approximate is enough.

Why: An image-prompt adapter injects a reference image's appearance through the model's attention layers, giving style transfer in seconds rather than the hours a LoRA costs, at the price of much less control over what exactly is transferred.

A An adapter transfers style through attention...: Fidelity is where a LoRA wins; the adapter's advantage is time, not accuracy.

C An adapter can be adjusted after...: A LoRA's strength is also adjustable at generation time, so this states a difference that does not exist.

D An adapter avoids the licensing question...: The rights position turns on the output and how closely it resembles the source, not on whether intermediate weights were created.

Lesson 41, p. 210

Four different seeds with pose and depth controls both at strength 1.0 produce nearly identical images. What does that indicate?

Answer C: Total constraint is too high, so almost no freedom is left for the seed to influence.

Why: Stacking conditioning signals removes the model's freedom in proportion to the total constraint, so the sum of the strengths matters more than any individual value and should stay near 1.2 to 1.5.

A The controls are contradicting each other...: Contradiction produces a specific damaged region that varies between seeds, not uniform agreement across all four.

B The seeds are too close together...: Sequential seed values produce entirely unrelated noise fields, so numeric closeness has no effect on similarity.

D The control images are being applied...: End-step timing does affect rigidity, but even a control running to completion leaves seed variation visible unless the total strength is already excessive.

Module 5 · Repair, extend, assemble

The module starts on p. 219.

MODULE 5 TEST

Module 5 test, Q1, p. 268

Inpainting a hand that occupies eighty pixels keeps producing the same mush. Which setting changes the outcome?

Answer A: Inpaint at full resolution, so the crop is regenerated at native size.

Why: Without that setting the masked region is regenerated at the whole image's scale, so a hand occupying eighty pixels gets eighty pixels of budget and cannot be drawn. With it on, the tool crops to the mask plus padding, regenerates at the model's native resolution and scales back in — the same model, eight hundred pixels of attention.

Module 5 test, Q2, p. 268

Why is grain always the last step in a repair sequence?

Answer D: Applied before a repair, it leaves the repaired patch with none.

Why: Each stage changes the pixels the next would otherwise have worked on, which is why the order runs from the changes that move the most pixels to the fewest. A repaired patch dropped into an already-grained image carries no grain and shows, so the global grade and grain happen on a flattened copy at the end.

Module 5 test, Q3, p. 268

Why are eyes masked together rather than repaired one at a time?

Answer B: They must agree on gaze, size and catchlight, which separate passes cannot.

Why: Eyes have to agree with each other — direction of gaze, relative size, where the catchlight sits — and a pass that sees only one has nothing to match against. Keep the denoise low as well, because above about 0.55 you are generating a different person's eyes.

Module 5 test, Q4, p. 269

Extending a portrait sideways by outpainting produces a second person. Why?

Answer C: The prompt still describes the person, and there is empty canvas.

Why: An extension is prompted for what continues in the new area, not for what the image is of. Describe the plain studio background and the soft falloff, and leave the description of the person out, or the model will fill the empty canvas with the thing you are still asking it for.

Module 5 test, Q5, p. 269

You are removing a cup from a table by inpainting. Which two adjustments make it work?

Answer D: Mask its shadow and surroundings too, and prompt for the empty table.

Why: A removed object that leaves its shadow is the most conspicuous possible failure, so the mask includes the shadow, any reflection and twenty to thirty pixels of surroundings. Negation is unreliable and naming the cup in any form raises the chance of getting one, so prompt for the background alone at a high denoise.

Module 5 test, Q6, p. 269

Why is a product label composited in after a tiled generative upscale rather than put through it?

Answer A: Tiles add plausible detail, so letterforms come back as different words.

Why: A generative upscale invents rather than restores, and it invents more the larger you print. Anything that must remain exactly itself — text, logos, product detail, a recognisable face — is kept out of the tiled pass and composited back in at full resolution afterwards.

THE LESSON CHECKS

Lesson 42, p. 223

Someone repeatedly inpaints a mangled hand in a 1024-pixel-wide image and it stays mangled, even though the same model draws perfect hands when asked for…

Answer D: Without "only masked" or "inpaint at full resolution" enabled, the region is regenerated at the whole image's scale, so the hand gets a few dozen pixels of budget and there is no room for the detail.

Why: Generate a base plate, then repair it region by region; the "inpaint at full resolution" setting is what gives a small detail enough pixels to be drawn correctly.

A The model is genuinely weak at…: Reaches for a model-level fix for a problem caused by how the region was rendered.

B The mask is feathered too heavily…: Names a real setting, but feathering affects the seam at the mask edge, not how much detail the region can hold.

C The prompt does not describe the…: Assumes description can compensate when the constraint is the number of pixels available.

Lesson 43, p. 229

Why must global grain and grading be applied after all masked repairs rather than before?

Answer B: Because a repaired region regenerates without the grain that was applied earlier, so the patch shows as a smooth area.

Why: Repair has a correct order — structure, then faces and hands, then edges, then small objects, then grade and grain last — because each stage changes the pixels the next stage would otherwise have worked on.

A Because grading before repair shifts the…: Plausible, but inpainting matches the surrounding pixels whatever their tone, so a graded image gives a correctly matched patch; the problem is texture, not colour.

C Because grading is destructive and cannot…: Describes poor file discipline rather than the reason for the order; a layered workflow makes grading non-destructive either way.

D Because the inpainting model is trained…: Training data covers heavily graded photographs throughout, and inpainting quality does not depend on whether a curve has been applied.

Lesson 44, p. 233

Why does "inpaint at full resolution" fix hands when ordinary inpainting at the same denoise does not?

Answer A: Because the cropped region is scaled up to the model's native size before denoising, so the hand is generated with far more pixels.

Why: Hands, eyes and teeth fail because they are small, high-variance structures rendered at too few pixels, so the fix is always to give the region more pixels rather than to describe it better.

B Because cropping to the masked region…: Context is reduced, but padding deliberately keeps some; removing context alone would not supply the resolution the structure needs.

C Because the setting increases the effective…: Steps are unchanged; the difference is spatial resolution, and more steps at 60 pixels still cannot encode a hand.

D Because a smaller region occupies less…: Invents an attention budget that works this way; the constraint is how many latent values describe the hand, not how attention is distributed.

Lesson 45, p. 237

Why is setting type in a separate tool the correct approach rather than a temporary workaround for current model limitations?

Answer C: Because typeface, weight, tracking and alignment cannot be expressed in a prompt, and a headline must be editable without regenerating the picture.

Why: Generated text fails at unusual words, small sizes and long strings because a diffusion model renders letterforms as visual texture rather than assembling glyphs, so any text that must be correct is set in a type tool over a clean plate.

A Because diffusion models will always render…: Overclaims permanence: short common words already render correctly on several models, and the argument for separation does not depend on failure rates.

B Because vector type scales to any…: A genuine advantage and not the main one; a raster headline at 300 dpi is perfectly adequate, while the inability to specify a typeface is not.

D Because generated text may accidentally reproduce…: A real hazard covered by the forbidden list, but it is a side issue rather than the structural reason for separating the two operations.

Lesson 46, p. 241

Why does outpainting in 256-pixel steps with overlap work better than one 512-pixel extension?

Answer B: Because each new pixel stays close to real pixels it can match light and perspective against, and the overlap hides the join inside regenerated area.

Why: Outpainting extends an image in overlapping steps of a few hundred pixels, and the overlap is what lets the model match light, perspective and texture rather than inventing a discontinuous new scene.

A Because a smaller masked area allows…: Denoise is already near 1.0 in both cases, since the area is empty; step size is not what permits a high value.

C Because the model processes the image…: Tiling is a different mechanism used for large canvases; ordinary outpainting is a single pass over the enlarged frame.

D Because repeated smaller passes allow the…: A genuine benefit of the method and a secondary one; the same prompt used throughout still gives a far better result in steps than in one leap.

Lesson 47, p. 245

An object is removed from a gravel path but leaves a faint smooth patch visible when the image is examined. What is the cause and…

Answer C: Texture was lost in regeneration, so the patch lacks the grain of its surroundings; add matching noise or run a low-denoise pass over it.

Why: Clone, heal, content-aware fill and inpainting fail in different ways, so the choice depends on whether the area behind the object is a texture, a structure or a scene that has to be invented.

A The mask was feathered too heavily…: Heavy feathering produces a soft halo rather than a smooth interior, and the fault described is in the middle of the patch rather than at its edge.

B The removal was performed with content-aware…: Content-aware fill is at its best on gravel; irregular natural texture is precisely the case it handles well.

D The denoise value was set too…: High denoise is correct for a removal and still matches surrounding texture through context; the smoothness comes from the regeneration, not from the value being wrong.

Lesson 48, p. 249

Which element most reliably gives away a composited product as not belonging to the scene?

Answer A: The absence of a tight contact shadow where the object meets the surface, which makes it appear to float.

Why: A photographed product composited into a generated scene is convincing only when perspective, light direction, contact shadow, reflection and grain all agree, and the contact shadow is the one that gives it away.

B A mismatch in resolution, because a…: Resolution mismatch is real and is corrected with blur and grain matching; a sharp product with a correct contact shadow still reads as present.

C An imperfect cutout edge, where remnants…: A visible fringe is an obvious fault and a local one that viewers read as a bad cutout rather than as the object not belonging in the space.

D A difference in colour temperature between…: Temperature mismatch is noticeable and easily corrected with one adjustment layer, whereas a missing contact shadow persists through every colour correction.

Lesson 49, p. 253

Why should grain be applied to a flattened composite rather than to each element separately?

Answer C: Because a single grain layer over everything is what makes separate sources read as one photograph.

Why: Every image carries a signature of noise, sharpness, depth of field and chromatic behaviour, and a composite reads as fake when two sources disagree on that signature rather than on content.

A Because per-element grain would require matching…: A practical difficulty and not the reason; even perfectly matched per-element grain leaves the elements reading as separate layers.

B Because applying grain to a layer…: Edge artefacts can occur with careless masking but are avoidable, and fixing them would still not make the sources cohere.

D Because grain applied before flattening is…: Transform order is a genuine technical caution, but grain applied per element after all transforms still fails to unify the image.

Lesson 50, p. 257

A cut-out subject shows a faint light line along its entire boundary, and the same line appears on other high-contrast edges elsewhere in the image…

Answer C: Sharpening or upscaling applied to the whole image, which raises contrast at every boundary and produces a light band on one side.

Why: A composite or repair fails at its boundary, and the three specific faults — colour contamination, a bright or dark halo, and over-feathering — each have a distinct cause and a distinct fix.

A Colour contamination from the original background…: Contamination takes the colour of the old background and appears only on the cut element, not on unrelated edges elsewhere in the frame.

B The mask was feathered too little…: Insufficient feathering gives a hard, jagged edge rather than a light band, and it would not affect edges the mask never touched.

D The composited layer's blend mode is…: A blend-mode fault is confined to the composited layer, so it cannot explain the same artefact on edges that were part of the original background.

Lesson 51, p. 261

Why must tiled diffusion run at a denoise value around 0.25 rather than 0.6?

Answer B: Because each tile is denoised independently, so at higher strengths a tile can invent content that contradicts its neighbours.

Why: A large final image is produced by generating at native size and enlarging in a separate pass, and where extra generated detail is needed the canvas is processed as overlapping tiles with a low denoise so no tile can invent a new scene.

A Because higher denoise values increase the…: Memory per tile is set by tile size rather than by denoise, and a 0.6 pass on a 768 tile fits wherever a 0.25 pass does.

C Because the overlap regions are blended…: Overlap divergence is a symptom of the same cause, and it would be solved by wider overlaps if it were the whole story, which it is not.

D Because tiled passes run after a…: A high denoise would replace those artefacts rather than amplify them; the problem is invention between tiles, not amplification within one.

Module 6 · The same thing, twelve times

The module starts on p. 271.

MODULE 6 TEST

Module 6 test, Q1, p. 314
Why will a fixed seed not give you the same character in a new pose?
Answer C: A seed reproduces one image, not an identity; one word changes the face.
Why: A seed fixes the starting noise. Same seed and same everything else gives the same picture bit for bit; change one word and the face changes with it. Seeds are indispensable for controlled tests and for returning to a frame you liked, and useless for carrying a person into a new scene.

Module 6 test, Q2, p. 314
Twenty training photographs were all taken in one session against one wall. What will the LoRA do?
Answer B: Treat that wall and that light as part of the person's identity.
Why: A LoRA learns whatever is constant across its training set. If the face, the lighting, the background and the angle are all constant, it learns them as one inseparable concept — so prompting for dramatic side light becomes a fight between the prompt and the LoRA, which the LoRA usually wins.

Module 6 test, Q3, p. 314
Your character LoRA always puts the subject in the same shirt. What went wrong in the captions?
Answer A: The shirt was not captioned, so it attached to the trigger token.
Why: Caption what should vary, not what should stay. Everything you name becomes separable and controllable afterwards; everything you leave uncaptioned attaches to the trigger token as part of the concept. It is also why a style LoRA's captions describe content thoroughly and never name the style.

Module 6 test, Q4, p. 315
You have four frames from an adapter at 0.75, side by side at delivery size. How do you decide whether to train a LoRA?
Answer D: Ask a stranger whether it is the same person, at delivery size.
Why: The question is not whether you can see the differences, having stared at them for days, but whether a viewer would, at the size the work is delivered and in the order they will meet it. Somebody who has not been in the project answers that immediately, and the answer is usually right.

Module 6 test, Q5, p. 315
Why is a product harder to hold consistent across a set than a face?
Answer C: Viewers compare it against the real thing, so any variation is a defect.
Why: A face varies with expression, angle and light, so a viewer's tolerance for small differences is wide. A manufactured object does not vary: its proportions, label and logo are fixed, and any difference between two images is a defect that can be checked against the item on a shelf.

Module 6 test, Q6, p. 315
One set has a slightly drifting face and one rigorously held look. Another has a perfect face and twelve different lighting setups. Which reads as…
Answer B: The first, because coherence is read from light, lens, palette and finish.
Why: A viewer asked whether twelve images belong together is not examining the nose. They are responding to whether the images look made by one person in one session, which is built from light, lens, palette, crop convention and finish — all cheap to enforce and all far more visible than small drift in a face.

THE LESSON CHECKS

Lesson 52, p. 275

A client wants their founder to appear in six generated scenes. The designer uses a single-photo face reference feature and the client says the results…

Answer C: Single-photo adapters transfer the broad geometry of a face and are family-resemblance accurate rather than identity accurate; a specific real person needs a LoRA trained on 20-30 varied photographs, or her real photographed head composited in.

Why: One-photo identity adapters get you a family resemblance; a specific real person needs a LoRA trained on varied photographs, or their actual photographed face composited in an editor.

A The reference photograph is too low-resolution…: Blames the input's quality for a ceiling that belongs to the method itself.

B The seed should be locked across…: Confuses reproducing one image exactly with carrying an identity across six different images.

D The prompt needs to describe her…: Expects a text description to carry identity information that the pixels themselves did not.

Lesson 53, p. 278

Why build a character sheet before generating any of the final frames?

Answer A: Because an identity cannot be described in words precisely enough, so it has to become an image you condition on.

Why: A character sheet turns an identity from something you hope recurs into an asset you condition on, and building it first is what makes every later frame a matter of reuse rather than of luck.

B Because generating the sheet first establishes…: Seeds do not carry identity; the same seed with a different prompt or resolution produces an unrelated person.

C Because a sheet built from the…: A real reason for keeping sheet images neutral, but it explains how to build the sheet rather than why it comes first.

D Because training a LoRA requires images…: Angles can be generated at any point, and many jobs never train a LoRA at all.

Lesson 54, p. 282

A character LoRA reproduces the grey wall from its training photographs no matter what environment the prompt asks for. What has happened?

Answer C: Overfitting: the wall was constant across the training set and never captioned, so it became part of the concept attached to the trigger token.

Why: A usable character LoRA trains from 15 to 30 images in under an hour on free cloud hardware, and the settings that matter are rank, learning rate and total steps, with overfitting visible as the training images reappearing in the output.

A The network rank is too high…: Excess rank can contribute, but the same symptom appears at rank 8 with too many steps, so rank is not the mechanism being described.

B The learning rate was set too…: A high rate accelerates the same outcome without being the cause; the background is learned because it was never separated from the concept.

D The trigger token chosen was not…: A non-rare token causes contamination from the model's existing meaning, not reproduction of a specific wall from your own photographs.

Lesson 55, p. 286

Why does a training set shot entirely in one lighting setup produce a LoRA that resists prompted lighting changes?

Answer C: Because a LoRA learns whatever is constant, so with lighting held fixed it becomes part of the concept rather than a separable attribute.

Why: A LoRA learns whatever is constant across its training images, so variety in everything except the subject is the property that decides whether it works, and no training setting compensates for a set that lacks it.

A Because the adapter's weights are optimised…: Frames the problem as a limitation of range rather than of association; the LoRA renders other tonal ranges perfectly well in regions it did not learn.

B Because lighting information is encoded in…: They share layers but remain separable in practice, which is exactly why a varied set produces a LoRA that takes prompted lighting.

D Because the captions described the subject…: Inverts the captioning rule: an uncaptioned constant is absorbed into the concept, not excluded from it.

Lesson 56, p. 290

When is an image-prompt adapter the right choice over a trained LoRA?

Answer B: When the character appears in one or two images seen separately, where drift between frames will never be observed.

Why: An adapter gives roughly seventy per cent of an identity in under a minute and a trained LoRA gives most of the rest in about an hour, so the decision turns on how many images the identity has to survive and how closely it will be examined.

A When the base model already renders…: Base-model competence does not reduce drift between frames; an approximate identity is approximate whether or not the model handles the category well.

C When the character must appear alongside…: LoRAs compose with controls routinely; the constraint is the total strength budget, which applies to adapters equally.

D When there are fewer than fifteen…: Fifteen is a practical floor for good results rather than a hard requirement, and for a fictional character the sheet can simply be generated.

Lesson 57, p. 294

Why is a product harder to keep consistent across a set than a human character?

Answer B: Because a product is fixed and checkable against the real item, so variation reads as a defect rather than as natural difference.

Why: A face is forgiven small variation and a product is not, because a viewer compares the rendered product against the real one they can buy, so compositing a photograph is almost always the correct answer.

A Because manufactured objects have hard edges…: Models handle both adequately; the difficulty is not in the rendering but in how closely the result can be checked.

C Because a product's label contains text…: The label is one of several problems and is solved by setting type; proportions and finish would still drift on a completely unlabelled object.

D Because character consistency techniques such as…: Object LoRAs train perfectly well; the constraint is the precision a viewer demands, not the availability of a method.

Lesson 58, p. 299

Why does holding five visual constants matter more to a set's coherence than perfect character consistency?

Answer B: Because viewers judge whether images were made together from light, lens, palette, crop and finish rather than from small facial differences.

Why: Coherence in a set is produced by holding light, lens, palette, crop and finish constant, and these are cheap to enforce and far more visible to a viewer than small drift in a character's features.

A Because character drift is only visible…: Display size is not the distinction; a mismatched frame stands out at any size and so does a substantially different face.

C Because the five constants can be…: Three of the five are generation-time decisions, and character drift can also be corrected afterwards by compositing.

D Because a set is usually viewed…: Sets are frequently seen side by side in grids, and the argument would reverse if they were not.

Lesson 59, p. 303

What is the practical test of whether a file naming and folder scheme is good enough?

Answer B: Whether the approved file for a given frame can be located in under thirty seconds several months later.

Why: A naming convention that encodes project, frame, version and status lets you find the approved file in seconds six months later, and its absence is what turns a small revision into an afternoon.

A Whether every file in the project…: Useful for reproduction and not the daily need; most revision requests are answered by locating a file, not by regenerating one.

C Whether the structure is consistent enough…: A worthwhile property, and handover is rare compared with the constant need to find your own approved files quickly.

D Whether every intermediate version has been…: Keeping everything makes retrieval harder rather than easier; the scheme works by making the important files findable, not by preserving all of them.

Lesson 60, p. 308

In a realistic twelve-image campaign, roughly what share of the total time is spent generating the non-anchor frames?

Answer C: Under ten per cent, because generation is fast once the specification, system, character and anchor are settled.

Why: A twelve-image campaign is about thirty hours of work distributed unevenly — a third on establishing the system, most of the rest on repair and consistency, and very little on generation itself.

A Around a third, since each of…: Confuses number of images produced with time spent; contact sheets run at seconds per frame and the candidates take minutes.

B Around a fifth, because generation and…: Bundles two very different activities: repair is indeed large, but it is separate from generation and roughly six times bigger.

D About half, since generation is the…: Repair, compositing and QA are also per-frame and take far longer; generation is the cheapest of the repeated stages.

Module 7 · Finishing and handing over

The module starts on p. 317.

MODULE 7 TEST

Module 7 test, Q1, p. 358

Which image should never go through a diffusion-based "creative" upscale?

Answer A: A packaging shot with a legible product label.

Why: Diffusion upscalers re-generate rather than sharpen, and what they invent is plausible rather than correct. Small text becomes different text with plausible letterforms, so anything carrying information — labels, logos, faces, diagrams — is enlarged conservatively or composited back at full resolution.

Module 7 test, Q2, p. 358

Why Curves rather than Brightness and Contrast on a raw generation?

Answer D: Curves lets you choose which tones move instead of shifting all of them.

Why: A raw generation sits in the midtones with weak blacks and no clean white. Pulling the bottom-left point right until the darkest shadow is genuinely black, the top-right left until something is genuinely white, then a gentle S in the middle, is twenty seconds and the single largest improvement available. A uniform shift crushes both ends instead.

Module 7 test, Q3, p. 358

An A1 poster will be read from about a metre. What resolution does it need?

Answer B: About 87 dpi by the arithmetic, so 150 with a safety margin.

Why: Required resolution is set by viewing distance: roughly 3438 divided by the distance in inches. The 300 dpi figure comes from the eye's resolving power at about thirty centimetres, which is how far away a magazine is held, and applying it to a poster produces seventy megapixels nobody can see.

Module 7 test, Q4, p. 359

A vivid cyan-blue sky flags grey under a CMYK soft proof. What should you do, and why then?

Answer C: Re-grade that range yourself now, while the image can still be changed.

Why: A screen emits light and a press absorbs it, so some colours cannot be printed at all rather than printed slightly differently. Letting the converter choose gives whatever the algorithm decides; re-grading toward colours the process can reach gives a controlled result, and the colour conversion is a decision rather than a checkbox at export.

Module 7 test, Q5, p. 359

A headline sits over a photograph. Where do you sample to test whether it is legible?

Answer D: At the brightest point behind light type, where it fails.

Why: Contrast fails at the extremes rather than on average, so the test is the worst point: the brightest pixel behind light type and the darkest behind dark type. If the worst point clears 4.5 to 1, the headline clears everywhere, and the check takes thirty seconds with any free contrast checker.

Module 7 test, Q6, p. 359

A client says the delivered file looks washed out on their machine. Which export setting is the likeliest cause?

Answer A: No colour profile was embedded, so the file was interpreted.

Why: A file without an embedded profile is interpreted by whatever opens it, and washed out or oversaturated on somebody else's machine is the usual symptom. Export sRGB with the profile embedded for screen, and the printer's CMYK profile embedded for print.

THE LESSON CHECKS

Lesson 61, p. 321

An illustrator enlarges a poster 4× using a creative diffusion upscaler and the small print in the corner comes back as different words in the…

Answer A: A diffusion upscaler re-generates each tile as a new image rather than sharpening it, so it produces plausible letterforms rather than the ones that were there; anything carrying information should be enlarged with a non-generative model, or set as real type afterwards.

Why: Diffusion-based upscalers re-generate rather than sharpen, so never point one at text, a logo, a product detail or a face you need to stay recognisable.

B The upscale model was trained mostly…: Blames the training domain for behaviour the method produces by design on any content.

C The text was too small in…: Assumes more input pixels would stop a generative step from re-inventing what it sees.

D The tiles overlapped too little, so…: Names a genuine tiling artefact, but one that misaligns detail across a seam rather than substituting correct words with different ones.

Lesson 62, p. 325

Six generated images for one brand page each look fine alone, but together they look like an assortment from a stock library rather than one…

Answer D: Every generation carries its own contrast and colour cast, so the set only reads as a set after a per-image curves and colour-balance pass matched to one chosen reference image.

Why: Curves for contrast, a colour pass to match the set, fine grain to fake a sensor, and real type over any generated words — that is the ten minutes that separates output from a deliverable.

A The prompts were not consistent enough…: Expects prompt wording to control the tonal and colour response, which is decided after generation.

B They were generated on different seeds…: Treats the seed as a style control rather than as the starting noise for one specific image.

C They need a style LoRA trained…: Reaches for training when a per-image tone and colour pass would resolve it in minutes.

Lesson 63, p. 329

A roll-up banner 850 mm wide will be read from about two metres. Why is 300 dpi the wrong target?

Answer B: Because the eye cannot resolve more than about 45 dpi at two metres, so the extra pixels are invisible to any viewer.

Why: Required resolution is set by viewing distance rather than by a universal figure, so a billboard needs far fewer pixels per inch than a brochure and most "high res please" requests are satisfied by a file you already have.

A Because large-format printers use a coarser…: A genuine constraint at the press and a second reason rather than the primary one; even on a fine screen the eye cannot resolve 300 dpi at two metres.

C Because a file of that size…: File size is an inconvenience rather than a reason; the point is that the detail is invisible, not that the software struggles.

D Because upscaling a generated image that…: Describes a consequence of pursuing the wrong target rather than why the target is wrong.

Lesson 64, p. 332

Why can a vivid cyan-blue on screen not be reproduced on a CMYK press?

Answer B: Because screens emit light while inks absorb it, and the reachable colour range of the two processes genuinely differs.

Why: Screens emit light and presses absorb it, so a set of vivid screen colours cannot be printed at all, and the conversion has to be seen and approved before delivery rather than discovered on the printed sheet.

A Because CMYK uses four inks rather…: Black is added for density and detail and is not applied to pure saturated areas; the limit is what the inks can reflect, not the number of plates.

C Because the conversion to CMYK is…: Bit depth affects banding in gradients rather than which colours are reachable; the same limit applies at 16 bits.

D Because print colour is measured under…: Viewing conditions do affect appearance, and the blue would remain unreachable even under bright light.

Lesson 65, p. 336

Light type over a photograph is legible across most of the frame but disappears over a bright cloud. What is the right first response?

Answer C: Move the type to the quietest part of the frame, or place a soft gradient scrim behind it.

Why: Type over a photograph fails on contrast rather than on typeface, and the fixes are all about creating a controlled surface for the words rather than about choosing better fonts.

A Add a hard outline to the…: The last-resort remedy applied first; a hard outline is the most visible amateur tell and there are five better options ahead of it.

B Increase the type size, since larger…: Larger text does have a lower threshold, but zero contrast against a bright cloud stays zero at any size.

D Convert the type to a darker…: Solves the cloud and breaks everywhere else, since the same dark type will now vanish into the darker regions the light type handled.

Lesson 66, p. 339

Why does flipping an image horizontally help at the end of a long session?

Answer B: Because your eye has adapted to the familiar version, and reversing it breaks that adaptation so structural faults become visible again.

Why: A written check performed at 100% on the final file catches the specific faults that generated images produce and that the person who made the image has stopped being able to see.

A Because it reveals composition faults that…: Reading direction does affect how a composition is scanned, and the flip is useful for faults that were always present rather than ones introduced by reversal.

C Because generated images frequently contain a…: A directional bias in the data would show as a tendency across many images, not as a fault detectable by flipping a single one.

D Because inpainted and composited regions were…: Seams are orientation-independent; what changes is your familiarity with the image, not any property of the joins.

Lesson 67, p. 344

An exported JPEG looks washed out on the client's machine but correct on yours. What is the most likely cause?

Answer C: The colour profile was not embedded, so the receiving application interpreted the values with a different profile.

Why: Export is where visible damage is introduced by compression, colour profile loss and resampling, and each has a specific setting that prevents it.

A The client's monitor is uncalibrated, which…: A real source of variation and not this one; an uncalibrated display shifts everything a viewer sees, not one file relative to the working document.

B The JPEG quality was set too…: Low quality produces blocking and mosquito artefacts around edges rather than a uniform reduction in saturation.

D The file was exported at 8…: Eight bits is correct for delivery and carries full contrast; reduced depth causes banding in gradients, not an overall washed appearance.

Lesson 68, p. 347

An image sits behind a headline purely as decoration. What is the correct alt text?

Answer C: An empty alt attribute, so screen readers skip it and do not interrupt the content with irrelevant description.

Why: Alt text describes what an image conveys in its context rather than listing what is in it, and whether to mention that an image is generated depends on whether that fact matters to understanding it.

A A brief description of the photograph…: Applies a rule that has an explicit exception; describing decoration interrupts the reader with content that carries no meaning.

B A description of the mood the…: Well intentioned, and it puts an interpretation rather than information in a field screen readers announce as content.

D A note that the image is…: Still announces the image and consumes the reader's time; an empty attribute removes it from the reading order entirely, which is the intended behaviour.

Lesson 69, p. 351

A client asks for the subject to face the other way. How should this be categorised and handled?

Answer C: As a structural change: flag the cost before starting, since it is a new frame with all the repair and compositing repeated.

Why: Revisions divide into adjustments, regional changes and structural changes, and the professional skill is saying which category a request falls into before agreeing to it.

A As a regional change, since only…: Reversing a subject changes how the light falls on them and how they sit against the background, so nothing downstream survives it.

B As an adjustment, because a horizontal…: A flip reverses the entire frame including the background, any text and the light direction, so it is not the same request.

D As a structural change that should…: Declining is not the professional response; the job is to state the cost and let the client decide whether the change is worth it.

Examination

The paper starts on p. 361.

Examination, Q1, p. 362
A 16 GB M2 MacBook against an 8 GB Nvidia desktop card, for SDXL work. What is the honest comparison?
Answer A: The Mac is three to five times slower but can load models the 8 GB card cannot.
Why: Apple Silicon works through Metal rather than CUDA and is typically three to five times slower than an equivalent Nvidia card. Unified memory is the compensation: a 16 GB MacBook can load models that will not fit an 8 GB desktop card, which makes it a real option rather than a consolation prize.

Examination, Q2, p. 362
What is the catch shared by Colab's free tier, Kaggle notebooks and Hugging Face Spaces for image work?
Answer D: Nothing persists, so you reinstall and re-download the model every session.
Why: All three are genuinely free and all three forget everything between sessions. A 23 GB checkpoint download can eat twenty minutes of your allowance before you have made a single picture, which is the real cost of the free route rather than any restriction on the models themselves.

Examination, Q3, p. 362
Why does the course insist on .safetensors rather than .ckpt?
Answer B: A .ckpt is a Python pickle, so loading it executes code from the file.
Why: Loading a pickle runs code contained in it, and malicious checkpoints have been distributed. A safetensors file is a plain tensor container that cannot execute anything. If only a .ckpt exists, convert it in an isolated environment or find another copy.

Examination, Q4, p. 363
You will run the same eleven-step process over image after image for a paying client. Which interface does the course recommend, and why?
Answer C: ComfyUI, because a saved graph does not decay the way notes do.
Why: A form-based interface stores a repeated process in your head and your notes, where it decays. A node graph makes the workflow a file you can save, version and re-run in eight months and get the same result, which is exactly what client work needs.

Examination, Q5, p. 363
On Windows with an Nvidia card, speed collapses after you load a second model and nothing errors. What is it?
Answer D: The driver is spilling VRAM into system RAM instead of raising an error.
Why: By default the Nvidia driver spills VRAM into system memory rather than raising an out-of-memory error, so generation does not crash — it becomes perhaps twenty times slower with no message. Turning off CUDA system memory fallback gives you a clean error instead of a mystery.

Examination, Q6, p. 363
On an 8 GB card, in what order should you attack the four consumers of memory?
Answer A: Offload the text encoder, tile the decode, cut the batch, then quantise.
Why: The text encoder runs once per prompt, so offloading it pays the transfer cost once rather than on every step. Tiling the decode removes the end-of-generation spike, halving the batch halves the activations, and rounding the weights that actually make the picture is the last resort rather than the first.

Examination, Q7, p. 364
How do you actually test whether a Q4 model costs you anything on your own work?

Answer C: Fix the seed, render at both, and compare small text and distant faces.

Why: Online quality claims about quantisation are unreliable because people compare renders at different seeds and describe a different picture as loss. With everything fixed, look at three places: small background text, the smallest face in frame, and thin repeated structure such as railings. That is where rounding error accumulates first.

Examination, Q8, p. 364
In a well-run job, where do nine out of ten renders happen, and why does it matter?

Answer B: At low resolution and low steps, where cost per render is lowest.

Why: Almost every compositional decision is visible at 512 pixels and twelve steps, and that render costs a fraction of a full-quality one. Making those decisions at final quality is like printing every draft of a document on photo paper, and it is the standard way a month of credits disappears with nothing usable to show.

Examination, Q9, p. 364
A 1280 × 720 thumbnail is usually seen at about 246 × 138. What follows for the design?

Answer A: Detail below a certain scale is wasted and contrast carries the frame.

Why: The file still has to be 1280 × 720, because that is what the platform wants, but the design decisions are made for the size it is displayed at. Fine detail is invisible there and contrast is everything, which changes what the frame should contain rather than what size it should be.

Examination, Q10, p. 365
A client needs a product cut-out for a shop listing, on transparency. What do you tell them?

Answer D: No diffusion model outputs transparency; generate on plain and cut out.

Why: A transparent PNG comes from generating on a plain background and cutting the subject out afterwards, which is ordinary editing work. Discovering this at delivery is an unpleasant hour, and it is one of the things the placement question at the start of a brief is there to surface.

Examination, Q11, p. 365
Why is the anti-reference worth more than two of the other five slots in a reference pack?

Answer C: It converts taste into a statement both sides can check later.

Why: An image that is close to the brief and wrong, with a sentence saying why, is the one reference that makes a disagreement visible now rather than at the review. The other five say what is wanted; this one says what would count as a miss, which is the part people otherwise discover too late.

Examination, Q12, p. 365
The hardest item on a forbidden list to catch is stereotype. Why can no prompt fix it reliably?

Answer B: The model produces the average its training data associates with your words.

Why: The model is not making an error; it is producing the average of what its training data associates with your words, and that average carries whatever the source material carried. Naming specifics rather than categories helps, and a person asking whether they would be comfortable if it were about them helps more. It is a review step, not a setting.

Examination, Q13, p. 366

You set the exact brand red in an editor and it still looks wrong in the frame. What is happening?

Answer A: Simultaneous contrast: the surrounding light shifts how the red reads.

Why: A pure brand red surrounded by warm generated light reads as orange-shifted; the same red on a cool background reads pink. This is a property of human vision rather than a colour error, which is why grading the whole scene toward the palette comes first and matching the specific element comes second.

Examination, Q14, p. 366

Which of these is the strongest candidate for photographing rather than generating?

Answer D: The dish a restaurant is actually selling.

Why: The test is whether the image makes a claim about a specific real thing. Food a business is selling is regulated in many markets and, beyond the law, a generated dish looks delicious and wrong — which sets up every customer for disappointment. The other three are stage dressing and generate well.

Examination, Q15, p. 366

Which request is a structural change rather than an adjustment?

Answer B: Can she face the other way?

Why: Turning the subject is a new frame, and everything downstream of it — repair, compositing, grade — is repeated. The other three are adjustments that are minutes of work if the file is layered. Saying which category a request falls into before agreeing to it is the professional skill.

Examination, Q16, p. 366

What does a vendor's training-data indemnity actually cover?

Answer C: Claims about what the model learned from, not what you prompted.

Why: Adobe, Getty and Shutterstock stand behind claims about their training data on business plans. Read the exclusions: it covers what the model learned from, and it does not cover you prompting a famous cartoon mouse. Ownership is a separate and unsettled question the vendor cannot resolve for you.

Examination, Q17, p. 367

The image is framed too tightly. Which slot do you change?

Answer D: Camera, because framing is a camera decision.

Why: This is the whole point of naming the slots. Without them, people respond to a framing problem by changing the subject description and then cannot explain why the image got worse. Shot size, focal length and camera height all live in the camera slot.

Examination, Q18, p. 367

How does prompt order differ between a CLIP-based model and a T5-based one?

Answer A: CLIP weights earlier tokens more; T5 is far less position-sensitive.

Why: CLIP-based models read in 75-token chunks and weight earlier tokens somewhat more heavily, so the subject goes early. T5-based models handle longer natural-language descriptions and are much less position-sensitive, and with those full sentences outperform comma-separated tags.

Examination, Q19, p. 367
What happens when you stack five camera terms that all point in the same direction?
Answer C: The distribution narrows to the most predictable image available.
Why: Every photographic term narrows the distribution toward the centre of what that term looked like in the training data. Five that agree — 85mm, f/1.4, golden hour, shallow depth of field, bokeh — all point at the same well-trodden place, and the result is the most conventional image the model can make.

Examination, Q20, p. 367
Why does naming the light also fix composition and mood as a side effect?
Answer B: Light and composition are not separable; a phrase decides both.
Why: "Backlit in a doorway" decides where the subject is, what the background does and how the eye moves through the frame. That is three slots' worth of control from one phrase, which is why light is the highest-leverage thing you can specify.

Examination, Q21, p. 368
What can a CLIP interrogator not do, and why?
Answer A: It cannot recover the picture, because it describes rather than inverts.
Why: An interrogator answers "what words fit this picture" by comparing the image against text embeddings. That is a different operation from recovering the noise, seed, model and prompt that produced it — which, for a photograph, never existed. The description is lossy in exactly the way descriptions always are.

Examination, Q22, p. 368
Your portrait's composition is right and the face is nearly there. What is the cheapest next move?
Answer D: Four variations at strength 0.15 around the same seed.
Why: A variation seed blends the starting noise between your seed and another, so at 0.1 to 0.3 you get the same composition with small differences. You already have the frame; you are asking for it again with the dice rolled slightly, and one of four will usually be materially better.

Examination, Q23, p. 368
You measured your checkpoint's CFG burn point at 8, then changed checkpoint. What now?
Answer C: Re-test, because the burn point is a property of the checkpoint.
Why: Parameters interact, and a one-at-a-time test tells you about that variable at your current settings. The burn point differs per checkpoint and is usually lower than people run, which is why the notes file has a block per checkpoint with a date on it.

Examination, Q24, p. 368
Why take two frames forward from a contact sheet rather than one?
Answer B: Some frames unravel under repair and you need somewhere to go.
Why: You mask the hands, the new hands are fine but the wrist reads oddly, you fix the wrist and the sleeve changes. Forty minutes later the image is worse than when you started. Save the untouched base render, and keep a second candidate so that the fallback is not another contact sheet.

Examination, Q25, p. 369

A tool gives you one box marked "reference image". Why does it matter which of three things it is doing?

Answer A: Structure, style and character references do completely different jobs.

Why: A structure reference keeps the layout and changes everything else. A style reference takes palette, brushwork and grade and applies them to a different subject. A character reference carries a face or object into a new scene, and it is the hardest and most disappointing of the three. Find out which before blaming your input.

Examination, Q26, p. 369

Why does a 400-pixel forward from a messaging app make a poor img2img reference?

Answer D: Compression blocks are reproduced faithfully as texture.

Why: At mid strength the model preserves what it is given, and blockiness is something to preserve. A cleaner reference is a stronger instruction, which is also why busy backgrounds should be cropped out and any text in the reference covered before it is uploaded.

Examination, Q27, p. 369

Somebody spends forty minutes raising denoise to move a subject to the right. What have they missed?

Answer B: By the time it can move, nothing else survives either.

Why: Img2img has no notion of objects; it denoises a whole latent toward a distribution consistent with the prompt, so a chair on the left stays on the left or becomes something else on the left. Img2img changes what things are made of and how they are lit. Controls change where things are.

Examination, Q28, p. 370

Canny conditioning on a compressed photograph gives thin phantom lines in the sky. Why, and what do you do?

Answer C: Compression blocks look like edges; raise the thresholds or use softedge.

Why: Canny finds edges, and JPEG blocks, paper grain and scanner noise all look like edges to it. Its threshold settings matter enormously — too low and you capture noise, too high and you lose the object — and a learned extractor such as softedge ignores texture entirely.

Examination, Q29, p. 370

A sketch-conditioned interior feels subtly impossible to everyone who sees it. What did the sketch not fix?

Answer D: The perspective, which the model followed faithfully including the error.

Why: Conditioning is faithful, including to your mistakes. If the drawn walls do not converge correctly the model reproduces that, and viewers notice without being able to name it. A sketch is for arrangement; a depth render from a blockout is for geometry, because software computes vanishing points perfectly.

Examination, Q30, p. 370

Your depth control is weak because a distant wall is visible through a window. What is the fix?

Answer A: Move the far clipping plane in so the range covers the scene.

Why: Depth is normalised across the whole visible range. With fifty metres in shot, the two metres that matter are squashed into a narrow band of grey and the control carries almost no information. Either move the far plane in or use Mist with a manually set start and depth.

Examination, Q31, p. 371
Two people shaking hands, each in distinct clothing. Why will regional prompting not solve it?
Answer C: The hands belong to both regions, so the split does not hold.
Why: Regional prompting works by removing the need to bind: each area gets its own conditioning and the descriptions never meet. Anything where subjects touch or overlap breaks that, so the answer is to generate them separately and composite, or to accept a repair pass.

Examination, Q32, p. 371
You run an image-prompt adapter at high weight and your prompt stops being followed. Why?
Answer B: Image features compete with the text for the model's attention.
Why: The adapter feeds the reference's features into the model's cross-attention layers alongside the text, so the model is attending to a picture and a sentence at once. If detailed prompt instructions suddenly stop landing, lower the adapter weight.

Examination, Q33, p. 371
An inpainted hand comes back pointing the wrong way. Which control is wrong?
Answer A: Padding, so the model cannot see the arm it belongs to.
Why: Padding decides how much surrounding context the crop includes. Too little and the model cannot see the arm the hand belongs to, so it draws a hand that makes sense on its own and not in the picture. Too much and you are back to a small pixel budget.

Examination, Q34, p. 371
After eight inpainting passes a region has drifted in colour away from the rest. What is the fix?
Answer D: Flatten and correct it with curves in a raster editor.
Why: Each pass is a fresh generation matched only approximately to its surroundings, so the region diverges a little every time. Fix it outside the model with a curves and colour-balance pass confined to that area, rather than trying to inpaint your way back to matching.

Examination, Q35, p. 372
A tool consistently modifies pixels outside your mask. What is going on?
Answer C: It masks the latent rather than the pixels, so edges bleed.
Why: Some implementations mask in latent space, where one latent cell covers several pixels, so the boundary bleeds. It is a property of the tool rather than of your mask, and switching tools is faster than fighting it.

Examination, Q36, p. 372
Why do teeth want a lower denoise than you would expect?
Answer B: High denoise produces a perfect even set that looks like dentures.
Why: The failure with teeth is usually irregularity rather than absence, so a small correction is what is wanted. Push the denoise up and you get the other classic failure — a uniform set that reads as dentures. Some asymmetry is what makes a mouth look real.

Examination, Q37, p. 372

Automatic face detection and repair is run over a twelve-frame set. What is the risk?

Answer A: It can shift the character's identity slightly between frames.

Why: Detectors apply the same denoise to every detection, so a face that was already fine is regenerated and may change. Across a set that is exactly the drift the consistency work is trying to prevent. Use it for volume, check every result, and turn it off when identity matters.

Examination, Q38, p. 372

What is the thirty-second test for a ghost left behind by an object removal?

Answer D: Duplicate the layer, apply strong curves contrast, and look.

Why: Ghosts that are invisible at normal contrast jump out under an exaggerated curve, whether they are a tonal mismatch, a texture mismatch or a rim of the original object's colour. The test takes half a minute and catches almost all of them before a client does.

Examination, Q39, p. 373

A dark-haired subject cut from a green background shows a green rim. Why is more feathering not the fix?

Answer B: Edge pixels genuinely contain a blend of both; shrink or decontaminate.

Why: At the boundary, individual pixels really do contain a mixture of subject and background, because that is what happened at the sensor. A binary mask cannot separate what is physically blended, so you decontaminate the edge colours, shrink the selection by a pixel, or paint it out.

Examination, Q40, p. 373

What prompt do you give a tiled diffusion upscale, and why?

Answer C: A short one naming medium and finish, because tiles see only themselves.

Why: Every tile gets the same prompt, so a tile containing only sky, handed a prompt about a woman in a sari, will try to put her there. Describe the medium and the finish — documentary photograph, fine grain, natural light — and let the low denoise refine what is already in each tile.

Examination, Q41, p. 373

What does a one-photograph identity adapter actually give you?

Answer D: A family resemblance: broad geometry, with texture and age drifting.

Why: These carry the proportions and general structure of a face and are family-resemblance accurate rather than identity accurate — the person's sister comes out. That is fine for a stylised character and not fine when the client is looking at a picture of themselves.

Examination, Q42, p. 374

Why is a character sheet shot in flat, neutral light on a plain background?

Answer A: It separates the identity from everything that will change later.

Why: Everything except the person is going to change in the final frames, so the sheet should carry the person and nothing else. The same principle explains dataset hygiene: a LoRA learns whatever stays constant, so anything constant besides the subject becomes part of the subject.

Examination, Q43, p. 374
The profile is the hardest angle to build from a front view. Why?
Answer C: There is least information about a profile in a front view.
Why: The adapter can only work from what the reference contains, and a front view says very little about the shape of a nose or a jaw from the side. Expect drift, expect eight attempts, and accept a close match — the sheet's job is to give a combined signal rather than a perfect one.

Examination, Q44, p. 374
You save every epoch during LoRA training. What is that for?
Answer B: So you can find the middle epoch, where identity and prompt both work.
Why: Generate the same prompt and seed with each saved epoch and lay them out as a grid. Early ones have weak identity and follow the prompt; late ones have strong identity and ignore it, with training backgrounds reappearing. The good one is in the middle, and it is almost never the last.

Examination, Q45, p. 374
Five frames from one burst sit in a training set of twenty. What does that do?
Answer A: The set is effectively smaller, and what they share counts five times.
Why: Near-duplicates count repeatedly toward whatever they have in common, so a set of forty near-identical images is a much smaller set than its file count suggests. Fifteen genuinely different images beat forty near-duplicates comfortably.

Examination, Q46, p. 375
A LoRA needs a strength of 1.0 before it shows up at all. What does that tell you?
Answer D: It is undertrained; you probably picked too early an epoch.
Why: A LoRA that needs a strength of 1.0 to register is undertrained, and one that dominates at 0.5 is overtrained. The working range is about 0.6 to 0.85, and when a LoRA sits outside it the epoch you chose from the saved set is usually the thing to revisit rather than the strength.

Examination, Q47, p. 375
You have one acceptable generated version of a concept product. What do you do next?
Answer C: Cut it out and composite it everywhere, as you would a photograph.
Why: The moment you have one acceptable version of the object, stop generating it and turn it into an asset. A cut-out is identical every time it is used, which is exactly the property a product needs and exactly the property no generation can guarantee.

Examination, Q48, p. 375
Repair and compositing took 35 per cent of a twelve-image campaign and generating the frames took 6. What does that imply about buying a faster…
Answer B: It barely moves the total; the work is before and after generation.
Why: Six hundred images were generated and twelve shipped, and almost all of the six hundred were low-step contact-sheet frames costing seconds each. The time is in specification, character work, repair, compositing and the consistency pass, none of which a faster card touches.

Examination, Q49, p. 375

Which class of upscaler is honest and never helps?

Answer A: Bicubic and Lanczos resampling.

Why: Resampling averages from what was already there: more pixels, no new information, bigger and softer. It never lies to you and it never helps, so it is the right tool when you need a specific pixel dimension and the detail is already sufficient.

Examination, Q50, p. 376

Why is upscaling always the last step?

Answer D: It bakes errors in at four times the size and multiplies later work.

Why: Fix defects at the base resolution, do the colour and contrast pass, and only then enlarge. If anything has to be re-run, every step after the upscale costs four times as much time and money, which is the practical reason rather than an aesthetic one.

Examination, Q51, p. 376

GFPGAN and CodeFormer restore facial detail well. What is the caution?

Answer B: At high fidelity settings they smooth skin into plastic.

Why: Face restoration models are trained to produce clean faces, and pushed hard they produce a waxy, poreless result that reads as fake in a different way from the original fault. Back the strength off until the skin still has texture.

Examination, Q52, p. 376

Why is serving a 6000-pixel hero image to a phone a real cost rather than harmless generosity?

Answer C: Page weight is paid by the people least able to afford it.

Why: Several megabytes of download for a four-hundred-pixel display is a cost borne by whoever is on the slowest connection and the tightest data plan. Two times the layout size covers high-density screens; beyond that there is no visible gain and a real cost.

Examination, Q53, p. 376

Why does 100 per cent K alone print as a washed dark grey over a large area?

Answer D: One ink cannot cover a large area densely; a rich black uses four.

Why: A single ink laid over a large solid area does not reach a convincing density. A rich black — around 60C 40M 40Y 100K, though printers vary — puts colour under the black to deepen it. Ask the printer for their preferred build rather than guessing.

Examination, Q54, p. 377

A headline over a busy photograph is failing contrast. Which remedy does the course rank last?

Answer A: A hard outline or drop shadow on the type.

Why: A tight, subtle shadow can work, but a hard outline reads as amateur work and is the most common mistake in this area. Moving the type is free and usually improves the composition; a gradient scrim is nearly invisible as a device and is what most well-designed posters do.

Examination, Q55, p. 377

An image sits purely as decoration behind a headline. What is the correct alt text?

Answer C: An empty alt attribute, so screen readers skip it.

Why: Alt text on a purely decorative image is noise: the reader hears a description of something irrelevant between them and the content. An empty alt attribute is the correct and accessible answer, and it is the case people get wrong most often.

Examination, Q56, p. 377

A client asks for a different expression on the subject. What category is that, and what does it cost?

Answer B: Regional; thirty to ninety minutes, and nearby areas may shift.

Why: One area regenerated or replaced is a regional change: slower than a grade, far cheaper than a new frame, and it may disturb what is around it so a joint pass often follows. Naming the category before starting is what keeps a revision from becoming a dispute.

MAKE THINGS

Image Generation, in Practice

Reference images, inpainting, character LoRAs, upscaling, and the licence nobody read.

WHAT IT TEACHES	The craft of getting a specific, finished image out of a model on a deadline. Choosing between hosted tools and open weights on cost, control and licensing.
WHAT IS INSIDE	Seven modules and 69 lessons, 27 figures drawn from data, seven case studies, seven module tests, a 56-question examination, and every answer with the reason behind it.
LICENCE	CC BY-NC-SA 4.0. Copy, print, share and adapt it for any non-commercial purpose, with credit, under the same licence. There is no paid edition.
THE COURSE	Free to read, with the lessons, the films and the examination, at **addaly.com/learn/ai-images**

Scan to open the course

Addaly

First edition, September 2026 · build 62a26075