MAKE THINGS

Video with AI

Plan the shots, generate ten takes, throw eight away, and cut what is left in a real editor.

First edition, September 2026

Addaly

Series editor: Dr Mustafa Mun

Draft, open for academic review

MAKE THINGS

Video with AI

Plan the shots, generate ten takes, throw eight away, and cut what is left in a real editor.

Series editor: Dr Mustafa Mun

Addaly

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

Video with AI

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). Video with AI. Addaly. addaly.com/learn/ai-video.*

This book is the printed form of the course of the same name at addaly.com/learn/ai-video. 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 · 66 lessons · 26 figures · 7 case studies · 51,153 words · about 10 hours

Brief contents

1

MODULE 1

What the machine is actually doing

Trace a prompt and a first frame through a latent video model — encoder, noise schedule, attention across time, decoder — and use that path to predict which settings change the picture, which change only the cost, why the final second degrades, and what VRAM a given model needs before you download it.

Lesson 1 · 8 min

Nobody is hiding the long version from you

THE ONE THING TO KEEP

Clip length is a memory-and-attention limit, not a product tier, so plan in shots and cut before the drift rather than trying to generate through it.

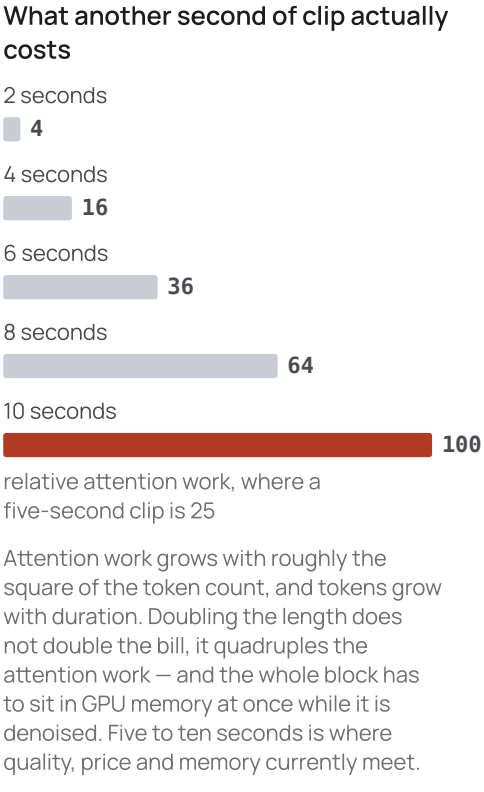
The limit is arithmetic, not marketing

Most people assume the few-second cap on generated clips is a pricing tactic — that a longer version exists behind a higher tier. It does not. The length falls out of how these models are built, and understanding why changes how you plan everything else.

A video model does not make frames one at a time. Frame 90 has to agree with frame 12 about where the wall is, so the model generates the whole clip as a single object. To make that affordable it compresses first: a video autoencoder squeezes the picture by roughly eight times in each spatial direction and by four times or more along time, turning a few hundred frames into a much smaller block of latent tokens. A transformer then denoises that entire block, attending across all of it.

Attention cost grows with roughly the square of the token count. Double the duration, roughly double the tokens, roughly quadruple the attention work — and the whole block has to sit in GPU memory at once while it is denoised. A ten-second clip is not twice the price of a five-second one. It is several times worse, and past a point it does not fit on the card at all.

Figure 1.1



That is the entire explanation. Five to ten seconds is where quality, price and memory currently meet.

Why the last second is the worst second

In image-to-video the first frame is given to the model. Every later frame is inferred, and each one is anchored by the frames before it rather than by anything fixed. Error accumulates with distance from the anchor.

You will see this in a consistent pattern:

- The first quarter-second is a settle. Motion starts slightly wrong and corrects.
- The middle is the good part.

- The last second is where the face loosens, the background churns, the colour creeps warm, and any object that left frame comes back subtly changed.

So generate long and deliver short. If you need six seconds on screen, generate ten. Treat the head and tail as offcuts, the way a printer treats bleed.

Extending compounds the damage

Every tool offers an extend button. What it does mechanically: takes the last frame of the previous generation — which is already a decoded, lossy output — and uses it as the conditioning image for a fresh generation. That frame gets re-encoded into latent space, generated from, decoded again.

This is a photocopy of a photocopy. Contrast climbs. Saturation creeps. Fine texture smooths into a painted look. Three extends in, the room is a different colour from where it started, and no prompt fixes it because nothing in the prompt changed.

If you must extend, do one of two things. Colour-match the sections back together in the edit afterwards. Or better, treat each extension as a separate shot and cut between them — a mismatch that reads as a cut is invisible, while the same mismatch inside a continuous move is glaring.

Plan in shots, and put the cut where the model is weakest

A three-minute explainer is thirty shots. Write it as thirty lines with a duration next to each one before you generate anything. This is not an AI workflow, it is a shot list, and it is the same document a crew would work from.

Then place your cuts deliberately. If the hand goes wrong at six seconds, the shot is five seconds long. You are not compromising; you are doing what editors have always done, which is end the shot before it stops working.

You can write this list on a phone in a notes app. It costs nothing and it is the highest-value thing in the whole pipeline.

When longer clips arrive, less changes than you expect

Models will keep stretching the window. It matters less than it sounds. Commercial editing runs at two to four seconds a shot; a sixty-second unbroken take is not something most projects want. The constraint you are fighting is already close to normal film grammar.

The thing a longer window does not fix is scenes. Two people talking across a cut, matching eyelines, the same jacket, the same key light — that is the unsolved problem, and it is about consistency between generations, not length within one. A later lesson deals with it honestly.

Today: take whatever you actually want to make, write it as a shot list with a duration on every line, and count the lines. That number, times your cost per shot, is your project.

BEFORE YOU MOVE ON

A team generates an eight-second clip and then presses extend three times to reach thirty seconds. The final section is noticeably more contrasty, more saturated and softer in detail than the opening. What happened?

A. Prompt adherence decays over long durations, so the later sections followed the description less closely than the first.

B. Each extension conditions on the last decoded frame of the previous generation, so encoding and decoding error compounds like a copy of a copy.

C. The clip passed the model's trained duration, so the tool fell back to frame interpolation to fill the remaining time.

D. Later generations were served from a lower-cost tier once the session's high-quality credits ran down.

The answer and why: p. 373

Lesson 2 · 10 min

Your clip is compressed fifty times before anything happens

THE ONE THING TO KEEP

A video autoencoder throws away roughly ninety-eight per cent of the pixel data before generation begins, so any detail finer than an eight-pixel block is being reinvented by the decoder rather than preserved.

The number that explains most of the artefacts

Take a five-second clip at 1080p and 24 frames per second. That is 120 frames, each 1920 by 1080 with three colour channels: about 746 million numbers.

No transformer attends over 746 million numbers. So the first thing a video model does is hand your frames to an autoencoder — usually called a 3D VAE or a causal video VAE — which squeezes them. A typical configuration compresses by 8× in width, 8× in height, 4× along time, and expands the channel count from 3 to 16.

Run the arithmetic. 120 frames become 30 latent frames. 1920×1080 becomes 240×135. With 16 channels that is $30 \times 240 \times 135 \times 16 \approx 15.5$ million numbers.

746 million in, 15.5 million out. Roughly forty-eight times smaller. Generation happens entirely in that small space. The decoder then expands it back to pixels at the end.

Figure 1.2

Where the 746 million numbers go

Pixels: 746 million numbers
A five-second 1080p clip at 24 frames per second. 120 frames, each 1920 by 1080, in three colour channels. No transformer attends over this.

Encoder: 8 by 8 by 4
Eight times narrower, eight times shorter, four times fewer frames, with channels rising from 3 to 16. Lossy, and lossy in a predictable direction.

Latent: 15.5 million numbers
30 latent frames at 240 by 135 by 16 channels. Roughly forty-eight times smaller. Every step of generation happens in here and nowhere else.

Decoder: back to pixels
Anything finer than about an eight-pixel block was never carried, so the decoder invents a plausible replacement: small type, fabric weave, a thin wire, an eyelash.

The squeeze is the reason the attention step is affordable at all, and it is also the reason generated footage has a slightly waxy surface and melting text. Neither was preserved and then damaged. Neither was ever in the latent.

What survives the squeeze, and what does not

The encoder is not a zip file. It is lossy, and it is lossy in a specific, predictable way: it keeps what it was trained to consider important and discards the rest.

What survives well:

- Large shapes, silhouettes, overall colour, the direction of light.
- Smooth gradients and broad textures.
- Motion of large objects.

What does not survive:

- Anything smaller than about an 8×8 pixel block at output resolution. That is a letter in small type, the weave of a fabric, a thin wire, an eyelash.
- High-frequency noise, including real camera grain.
- Fine specular highlights — the tiny bright points on wet surfaces, metal edges, eyes.

When you see those things in the output, they were not preserved. The decoder invented plausible replacements. That is why generated footage has a slightly waxy, denoised surface even when nothing has obviously gone wrong. It is also why text melts: the letterform was never in the latent to begin with.

The round-trip test you can run for free

This is the single most convincing demonstration in the subject, and it costs nothing because it involves no generation at all.

In ComfyUI, build a three-node graph: load a real video, encode it with the model's VAE, decode it straight back out. No diffusion, no prompt, no sampling. Compare input and output.

```
LoadVideo -> VAEEncode -> VAEDecode -> SaveVideo
```

What comes back is the absolute ceiling on quality for that model. Nothing generated through that VAE will ever be sharper than your round-tripped real footage. Do it once with a clip containing a sign, a patterned shirt and a face, and you will stop blaming the sampler for things the encoder did.

If you cannot run ComfyUI, several Hugging Face Spaces expose the same encode-decode path in a browser.

Causal encoders and why the first frame is different

Most current video VAEs are causal: latent frame n is computed from pixel frames at or before it, never from later ones. The practical consequence is that the very first frame is encoded on its own, without temporal compression,

while every later group of four frames is squeezed together.

That asymmetry is part of why image-to-video works as well as it does. Your supplied still enters the latent space with less loss than any frame that follows it. It is the sharpest frame in the clip before generation even starts.

It is also why the first quarter-second often looks subtly different from the rest — the settle described in the previous lesson has an encoder component as well as an attention component.

What you do differently knowing this

1. **Do not put load-bearing fine detail in the still.** If the logo has to be legible, it goes on in the editor as real text over the top, not in the frame you animate.
2. **Do not send a 4000-pixel still into a 720p model.** It is downscaled on the way in. You paid for pixels that were deleted before the model saw them.
3. **Do not expect an upscaler to recover what the VAE removed.** Upscaling invents new detail; it cannot restore the specific detail that was discarded. It is a finishing step, not a repair.
4. **Add grain at the end, deliberately.** The encoder removed real grain. A uniform grain pass over the finished timeline replaces a texture the pipeline is structurally unable to keep, and it is why generated footage that has been grained reads as film while the raw export reads as CGI.
5. **Judge a model's VAE separately from its sampler.** Two models with the same sampler quality can differ enormously in how much the decoder smears. The round-trip test isolates it.

The honest limitation

Bigger latents would help. Some newer models compress less aggressively along time, or use more channels, and they look visibly crisper. They also cost proportionally more to generate, because everything downstream scales with

the size of the latent. The waxiness is not a bug somebody forgot to fix — it is the price paid to make the attention step affordable at all, and it moves only when somebody accepts a bigger bill.

Today: run the encode-decode round trip on ten seconds of your own phone footage. Whatever comes back is the best this model will ever give you.

BEFORE YOU MOVE ON

You generate a shot containing a shop sign. The letters shimmer and reshape across the clip. You raise the sampler steps from 20 to 50 and regenerate. The letters still shimmer. Why?

A. Fifty steps is still too few for text; letterforms typically need 80 to 120 steps before they stabilise.

B. The letters are below the resolution the autoencoder preserves, so the decoder reinvents them each frame however well the latent was denoised.

C. The prompt did not specify the text clearly enough, so the model had no target to converge on.

D. The temporal attention window is shorter than the clip, so the letters lose their anchor to the first frame partway through and begin drifting away from it.

The answer and why: p. 374

CASE STUDY · MODULE 1

The twenty-second shot the venue insisted on

An illustrative case, built from documented patterns.

A two-person wedding and events film unit in Nashik, hired by a banquet venue that wanted one unbroken twenty-second aerial-style reveal of its lawn for the top of its website.

Rehan and Sneha had shot the venue on the ground twice. They had no drone permission for the site and no budget to hire one, so the plan was a generated reveal: start tight on the marigold arch, lift, and pull back to show the lawn, the lit mandap and the hills behind.

The brief said one continuous take. The venue's owner was explicit about it. He had seen a competitor's website with a single flowing shot and he believed cuts would make his lawn look smaller, which is not an unreasonable instinct for a man selling space.

The model gave them five-second clips. They tried the obvious thing first and pressed extend three times.

What came back was a twenty-second file that got worse as it went. The first five seconds were the shot they had paid attention to. By the third extension the marigolds had gone orange-red, the lawn had warmed by what Sneha measured as a noticeable shift on the parade scope, and the fine lattice on the mandap had smoothed into something painted. Nothing in the prompt had changed. Nothing in the settings had changed.

Rehan understood why once he thought about what extend does mechanically. The button takes the last frame of the previous generation — a frame that has already been decoded out of the latent and is therefore already lossy — and re-encodes it as the conditioning image for a fresh run. Three extensions is a photocopy of a photocopy of a photocopy. Contrast climbs, saturation creeps, high-frequency texture dies. There was no prompt that could have prevented it because the damage was not in the prompt.

That left a decision with a cost on each side.

They could deliver the twenty-second take and colour-match the four sections back together in the edit. That is a real technique and it works. But the join sits inside a continuous camera move, which is the worst place for a mismatch to live, because the eye has nothing else to attend to and any residual difference reads as a glitch rather than as a change of shot. Sneha estimated an evening of secondaries, and even then the lattice would still be soft in the last third.

Or they could cut. Four shots, four to five seconds each: the arch, the lift, the lawn, the hills. Each one generated fresh, each one a clean first frame, each one ending before the drift. That would cost them a conversation with a client who had already said no to cuts.

They chose to have the conversation, and they prepared for it rather than arguing. They built both versions. The extended one was ready first. Then they built the cut version and put a single grade and a light grain across all four shots, with a slow push on each so the movement carried through the joins, and laid the venue's own ambience — recorded on a phone at the site, birds and a generator in the distance — continuously underneath so the audio did not break at the cuts.

Then they sent both, unlabelled as to which was which, and asked the owner which looked more expensive.

The honest part of this story is that they were not certain he would pick theirs. The cost of being wrong was a re-edit they would not be paid for, and a client who felt overruled.

YOUR ANSWERS

1. Why did the colour and texture degrade across the three extensions when nothing in the prompt or the settings changed?

2. The team could have colour-matched the four extended sections together in the edit. Why is cutting between them a stronger answer than matching them?

3. What did generating four shots instead of one long take cost them, and was it worth it?

STOP HERE

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

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

CASE STUDY · MODULE 1

How it played out

The owner picked the cut version without hesitation and said it looked like it had been shot properly, which was the word he used. He had never objected to cuts as such; he had objected to a piece that felt small, and four well-graded shots with continuous sound felt larger than one drifting take. Rehan wrote three lines in his notes file afterwards: never more than one extension, generate ten seconds to deliver six, and show two versions rather than argue about one. The venue booked them for a second film in the same month.

The questions, discussed

1. Why did the colour and texture degrade across the three extensions when nothing in the prompt or the settings changed?

Extending conditions a new generation on the last decoded frame of the previous one. That frame has already been through the autoencoder once, so it has lost fine texture and picked up whatever bias the decoder has. Re-encoding it feeds that loss back in as the starting point, and the loss compounds with each pass. It is a photocopy of a photocopy: contrast climbs, saturation creeps, and high-frequency detail like the mandap lattice smooths into a painted look. No prompt can address it because the damage happens in the encode-decode round trip, not in the text conditioning.

2. The team could have colour-matched the four extended sections together in the edit. Why is cutting between them a stronger answer than matching them?

A residual mismatch inside a continuous camera move is glaring, because the eye is tracking one uninterrupted motion and any change in colour or texture arrives as a fault. The same mismatch either side of a cut is largely invisible, because a cut is already a change and the viewer expects the picture to be different. Cutting also lets each shot start from a fresh, sharp first frame rather than a degraded one, so the texture problem is prevented rather than corrected.

3. What did generating four shots instead of one long take cost them, and was it worth it?

It cost them a difficult conversation with a client who had specified a continuous take, plus the work of building two versions rather than one, plus four first frames instead of one. It was worth it because the alternative was an evening of secondary colour work on a shot that would still have had a soft final third, and because the cut version could be assembled from clips that each ended before the drift. The general lesson is to plan in shots and put the cut where the model is weakest, which is also ordinary film grammar.

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

1 A progress bar shows coloured fog rather than the first second of your clip. What does that tell you about how the model works?

A. There is no first second yet: the whole block is denoised at once
B. The frames are generated in order, but the decoder holds them back until colour matching finishes
C. Frame interpolation has to complete before any frame can be displayed
D. The provider buffers the file until the whole generation has been paid for

2 In image-to-video, why is the final second of a clip reliably the worst second?

A. Interpolation runs out of reference frames before it reaches the end
B. The model lowers its quality near the end to save compute
C. Error grows with distance from the one fixed anchor, the supplied first frame
D. The autoencoder compresses frames near the end of the clip more aggressively than those at the start

3 A generated clip contains a shop sign whose letters change shape every frame. What does that tell you about the pipeline?

A. The prompt did not name a font, so the model picked a different one each time
B. The sampler needs more steps to resolve the letterforms
C. Guidance was too low for the text instruction to be obeyed
D. The letters were never in the latent; the decoder reinvents them each frame

4 The model ignored your instruction that the woman turns left. You raise guidance from 5 to 9. The most likely result is that the clip

A. shows her turning left, at the cost of slightly stronger contrast in the highlights

B. still does not show her turning left, and is stiffer with cooked colour

C. is unchanged, because guidance only affects the cost of the run

D. runs at a higher step count, which the interface sets automatically

5 You saved the seed, the prompt and every setting for a shot generated on a hosted service. Six weeks later the same call returns a different clip. The most likely explanation is that

A. a seed fixes only the first frame's noise, and the rest is drawn fresh every run

B. seeds expire after a fixed period on hosted services

C. the provider batched, rerouted or quietly updated the model

D. the text encoder normalises prompts differently on each request

6 A 14-billion-parameter model at fp8 is about 14GB. Why is a 14GB card not enough to run it?

A. Activations and the VAE decode have to fit as well, and the decode spikes highest

B. The operating system reserves half of any card's memory for the display

C. The text encoder is larger than the video transformer and has to stay resident for the whole run

D. fp8 weights are expanded back to bf16 when they are loaded

Lesson 11 · 9 min

The still frame is where the decisions belong

THE ONE THING TO KEEP

Every decision you can make in a still costs a hundredth of what it costs to make in a clip, so fix the frame before you ever spend on motion.

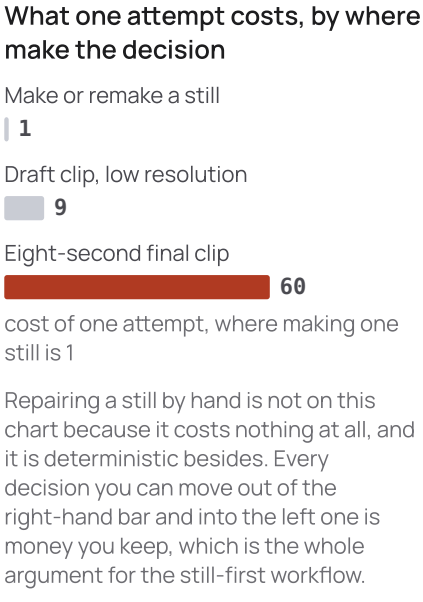
Two ways in, and they are not equal

Text-to-video: you type a description and the model invents the framing, the faces, the wardrobe, the colour, the light and the lens, then animates its own invention. You are gambling on a dozen decisions simultaneously, at video prices.

Image-to-video: you supply the first frame. The model's only job is motion.

Almost everyone doing this professionally works the second way, and the reason is economic before it is aesthetic. A still costs a cent or two and takes seconds. An eight-second clip costs fifty to a hundred times that and takes a minute or two in a queue. Image-to-video moves every hard decision out of the expensive loop and into the cheap one.

Figure 2.1



There is a second reason that matters more. A still is not only cheaper to regenerate — it can be repaired by hand. You can photograph it. You can composite the client's real product into it in Photopea, which runs in a browser tab, opens PSD files and needs no install. You can fix a bad hand in GIMP or Krita. You cannot do any of that inside a video model. Whatever you fix in the still is fixed for the whole clip.

For making the still, see ai-images. For repairing it, image-editing.

The first frame is a contract

Everything in that frame persists and then degrades. Six fingers in the still means six fingers for eight seconds, getting stranger. A wobbly sign in the still becomes a melting sign. A face you are not quite happy with in the still is a face you will hate by second five.

The frame also fixes the style. The model will not restyle mid-clip — it animates what it is given. So the still is where you decide whether this is a photograph, a gouache painting or a 3D render, and that decision is then locked in a way no prompt word can override.

EXAMINATION

Video with AI — final examination

This paper covers the whole course:

- how a latent video model turns a prompt and a first frame into a clip
- what belongs in the still rather than in the prompt
- how to write a shot instruction a model can act on
- what it costs to hold a character and a location across a sequence
- the audio half of the job and the consent and labelling that go with it
- the edit that turns generations into a piece
- how to plan, price and record a paid job

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 418.

1 Module 1 · What the machine is actually doing

Why is a ten-second clip considerably more than twice the price of a five-second one?

A. Longer generations are billed at a premium rate by every provider
B. Longer clips are placed in a lower-priority queue, and skipping that queue costs extra credits
C. Attention work rises with the square of the token count, and the block must fit at once
D. The autoencoder has to be run twice for clips over eight seconds

2 Module 1 · What the machine is actually doing

After three uses of the extend button, the room is a different colour and the fine texture has gone flat. What is the mechanism?

A. Each extension re-encodes an already-decoded frame, so the loss compounds
B. Extensions are generated at a lower resolution to keep the price down
C. The model reuses the same seed and drifts toward its unconditional prediction
D. The prompt's influence decays across successive generations

3 Module 1 · What the machine is actually doing

Most current video autoencoders are causal. What practical consequence does that have for image-to-video?

A. The last frame is encoded without temporal compression instead of the first
B. Motion can only run forwards, so reverse camera moves have to be flipped in the edit
C. The clip can be streamed out as it is generated, one frame at a time
D. Your supplied still enters the latent with less loss than any frame after it

4 Module 1 · What the machine is actually doing

A person walks behind a pillar and emerges wearing a different shirt. No setting prevents this. Why not?

A. The occlusion mask is recalculated at each denoising step

B. Nothing in the model holds an object′s identity; the tokens carrying it stopped existing

C. The attention window is too small to span the width of the pillar and the frames either side

D. The prompt did not specify the shirt in enough detail for it to persist

5 Module 1 · What the machine is actually doing

Why does adding a fourth clause to a video prompt so often change nothing at all?

A. A few hundred text tokens are steering a quarter of a million video tokens

B. Cross-attention layers are switched off after the first half of the denoising run

C. The extra clause is added to the negative branch by default in most tools

D. Most interfaces truncate prompts after three clauses

6 Module 1 · What the machine is actually doing

Asking a model for "a montage, then it cuts to the street" returns a morph or a camera move. What in the training explains that?

A. Cuts are filtered out at inference by the safety layer in most hosted tools

B. Aesthetic scoring removed clips containing hard cuts

C. Scene detection split the training data into single unbroken shots

D. Captioners were instructed not to describe edits

Module 1 · What the machine is actually doing

The module starts on p. 9.

MODULE 1 TEST

Module 1 test, Q1, p. 60

A progress bar shows coloured fog rather than the first second of your clip. What does that tell you about how the model works?

Answer A: There is no first second yet: the whole block is denoised at once

Why: Generation starts with the entire latent block full of noise and refines every latent frame together, step by step. At step seven, the first frame and the last frame are both seven steps along. A preview is a partially denoised block being decoded, which is why it looks like fog. The same fact explains why you cannot add a bit more onto the end and why the cost is fixed before any picture exists.

Module 1 test, Q2, p. 60

In image-to-video, why is the final second of a clip reliably the worst second?

Answer C: Error grows with distance from the one fixed anchor, the supplied first frame

Why: The first frame is given to the model. Every later frame is inferred, anchored only by the frames before it rather than by anything fixed, so error accumulates with distance from that anchor. The practical response is to generate long and deliver short: treat the first quarter-second and the last second as offcuts, the way a printer treats bleed.

Module 1 test, Q3, p. 60

A generated clip contains a shop sign whose letters change shape every frame. What does that tell you about the pipeline?

Answer D: The letters were never in the latent; the decoder reinvents them each frame

Why: The autoencoder compresses by roughly eight times in each spatial direction, so anything finer than about an eight-pixel block at output resolution is not carried. A letterform is exactly that fine. The decoder produces a plausible replacement independently for each frame, which is why text melts. More steps sharpen what the model decided to depict; they cannot restore information the encoder never kept. Text goes on afterwards, in the editor.

Module 1 test, Q4, p. 61

The model ignored your instruction that the woman turns left. You raise guidance from 5 to 9. The most likely result is that the clip

Answer B: still does not show her turning left, and is stiffer with cooked colour

Why: Guidance extrapolates past the conditioned prediction, away from the unconditional one. Pushing harder drives every frame toward the strongest single interpretation of the prompt, and the strongest interpretation does not change over time, so motion freezes while contrast and saturation climb. Guidance cannot supply an instruction the model did not act on. The fix is the first frame or the phrasing.

Module 1 test, Q5, p. 61

You saved the seed, the prompt and every setting for a shot generated on a hosted service. Six weeks later the same call returns a...

Answer C: the provider batched, rerouted or quietly updated the model

Why: A seed fixes the starting noise and nothing else. Reproducibility also requires identical weights down to the checkpoint, identical precision, step count, scheduler, tokenisation, batch shape and GPU kernels. A hosted endpoint batches with other requests, may route to a different GPU generation and may update the model, and providers generally reserve the right to all three. This is why the output file itself is the primary record, not the seed.

Module 1 test, Q6, p. 61

A 14-billion-parameter model at fp8 is about 14GB. Why is a 14GB card not enough to run it?

Answer A: Activations and the VAE decode have to fit as well, and the decode spikes highest

Why: Three things occupy VRAM at once: the weights, the activations the attention layers hold during each forward pass, and the full-size frames allocated when the latent is decoded back to pixels. The decode of a 1080p clip in one go can peak higher than the generation did. Leave three or four gigabytes over the weights, or use a tiled decode, which removes almost all of that spike.

THE LESSON CHECKS

Lesson 1, p. 14

A team generates an eight-second clip and then presses extend three times to reach thirty seconds. The final section is noticeably more contrasty, more saturated...

Answer B: Each extension conditions on the last decoded frame of the previous generation, so encoding and decoding error compounds like a copy of a copy.

Why: Clip length is a memory-and-attention limit, not a product tier, so plan in shots and cut before the drift rather than trying to generate through it.

A Prompt adherence decays over long durations...: Attributes a compounding image-quality loss to instruction-following, when the prompt is re-applied identically at each step.

C The clip passed the model's trained...: Invents a fallback mechanism and confuses a training window with a different technique entirely.

D Later generations were served from a...: Reaches for a billing behaviour where a signal-processing mechanism explains the result.

Lesson 2, p. 19

You generate a shot containing a shop sign. The letters shimmer and reshape across the clip. You raise the sampler steps from 20 to 50…

Answer B: The letters are below the resolution the autoencoder preserves, so the decoder reinvents them each frame however well the latent was denoised.

Why: A video autoencoder throws away roughly ninety-eight per cent of the pixel data before generation begins, so any detail finer than an eight-pixel block is being reinvented by the decoder rather than preserved.

A Fifty steps is still too few…: Treats a representational limit as a convergence problem, and invents a step count at which a discarded detail reappears.

C The prompt did not specify the…: Assumes the failure is in conditioning rather than in the pixel budget; a perfectly specified sign fails identically.

D The temporal attention window is shorter…: Names a real mechanism that causes other drift, but text shimmers even inside a single attention window because the detail was never in the latent.

Lesson 3, p. 24

A studio wants to show a client the first two seconds of a ten-second generation while the rest finishes rendering, to save waiting time. Why…

Answer D: Every frame is at the same stage of denoising throughout, so before the last step the first two seconds are as unfinished as the rest.

Why: A video model denoises every frame of the clip simultaneously across a fixed number of steps, which is why nothing can be streamed out early, why step count hits diminishing returns fast, and why distilled turbo models trade motion range for speed.

A The API does not expose partial…: Blames the interface for an architectural property; a local run has the same block and the same problem.

B Early frames are rendered at lower…: Invents a progressive per-frame refinement order that flow-matching video diffusion does not use.

C The decoder can only run on…: Half right about the decoder but wrong about the latent: no group of latent frames finishes ahead of the others.

MAKE THINGS

Video with AI

Plan the shots, generate ten takes, throw eight away, and cut what is left in a real editor.

WHAT IT TEACHES	Generative video taught as production work rather than a demo reel.
WHAT IS INSIDE	Seven modules and 66 lessons, 26 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-video**

Scan to open the course

Addaly

First edition, September 2026 · build 62a26075