

Inside the generator

Why does it produce this?



Harry Yang
15 September 2026
75-minute lecture

Three questions for today

Why does the same prompt give different pictures?

Why can stronger guidance make a picture worse?

Which setting should we change for the result we want?

Content, layout, and appearance

Part of the image	Example instruction
Content	One person, one coat, one suitcase
Layout	Small figure on the left; open space on the right
Appearance	Soft edges and a limited color palette
Intended use	Animation frame, poster, or installation image

One prompt can leave some of these choices unspecified.

One prompt can produce many pictures



Published SD v1-4 outputs using the same prompt: "a photograph of an astronaut riding a horse"

Recognizing an image and making an image

A classifier might label a picture “person at a station.”

A generator must choose the person, viewpoint, light, and background.

The prompt leaves choices open

“A traveler at a station” does not say where the camera should be.

It also leaves the pose, light, and colors open.

The model chooses one of many possible pictures.

How words and reference images help

Words can describe what we want to see.

A reference image can show where things should go.

The model uses these inputs, but may still miss details.

These extra inputs are called conditions.

Camera distance changes perspective

A Far away



B Eye level



C Looking up



Illustrations made for this class. Framing, pose, and light vary across the pictures.

Moving the camera changes perspective. Cropping changes framing.

Three depth cues for a flat picture

Cue

Overlap: one shape blocks another.

Relative size: repeated objects get smaller.

Station example

Place the traveler in front of a bench.

Let roof columns shrink toward the distance.

Converging lines add another cue: rails meet toward a vanishing point.

What to save with an image

Save this	Why it matters
Prompt and reference images	What you asked the model to make
Model and any LoRA	Which learned weights you used
Seed	How the random numbers started
Sampler and steps	How the image changed during generation
Guidance and image size	Other settings that can change the result

Learning and making an image

Training: learning

Use many examples.

Change the model's weights.

Sampling: making one image

Use the trained model.

Change the noisy image, usually keeping weights fixed.

More sampling steps do not train the model again.

What happens when we add noise?



Forward noising of one image. Source: The Annotated Diffusion Model (2022).

Clean image plus noise

$$\mathbf{x}_t = \mathbf{a}_t \mathbf{x}_0 + \mathbf{b}_t \boldsymbol{\varepsilon}$$

$\mathbf{x}_0$: the clean image

$\boldsymbol{\varepsilon}$: random noise

$\mathbf{a}_t$ and $\mathbf{b}_t$: how much image and noise to mix

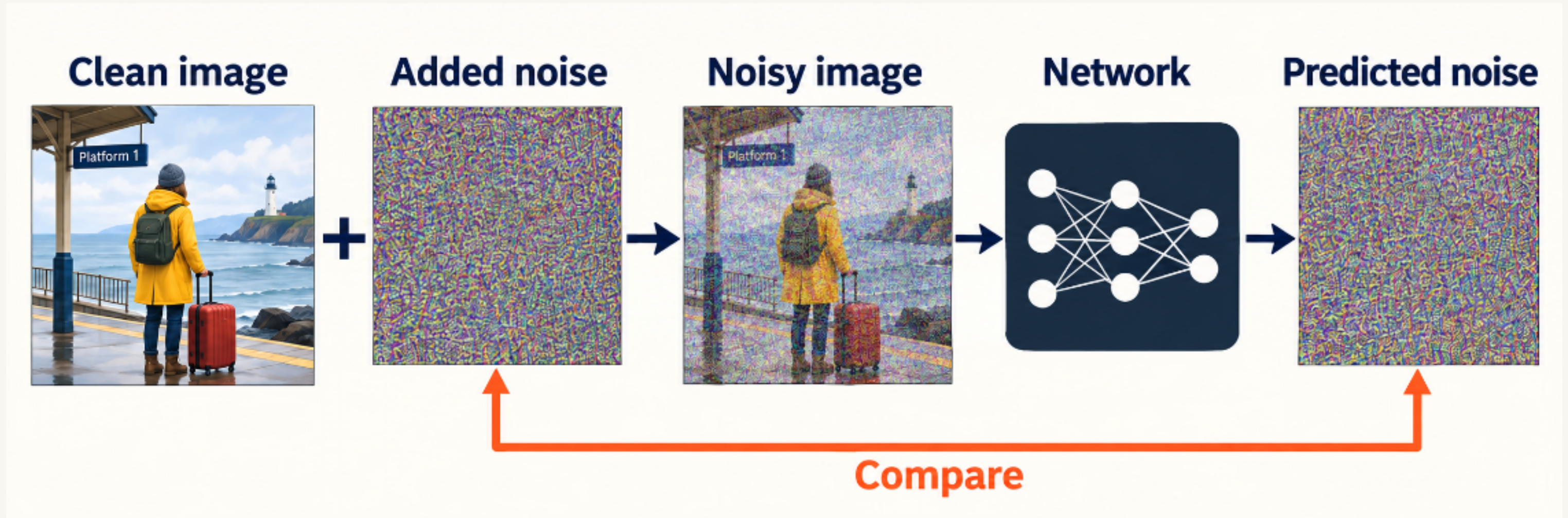
We can choose how noisy a training image is

Choose an image and a noise level.

Add a known amount of random noise.

The noisy training example is ready.

How the model learns to predict noise



We add the noise ourselves, so we know the answer.

The model tries to predict that noise.

Simplified training example. The noise pictures are illustrations.

How we measure a prediction error

$$\text{loss} = \text{average}[(\epsilon - \text{predicted } \epsilon)^2]$$

Compare the added noise with the predicted noise.

Larger differences receive a larger penalty.

Average the errors across many training examples.

This error score does not directly judge composition or mood.

Where does the finished image come from?

At high noise, many different clean images could fit.

The model uses patterns it learned from training images.

During generation, there is no original picture hidden in the noise.

Discussion: training or sampling?

Training on a new image set: which values change?

Generating with a new seed: which values change?

Which action can change learned subject behavior?

Distinguish learned weights from the state being generated.

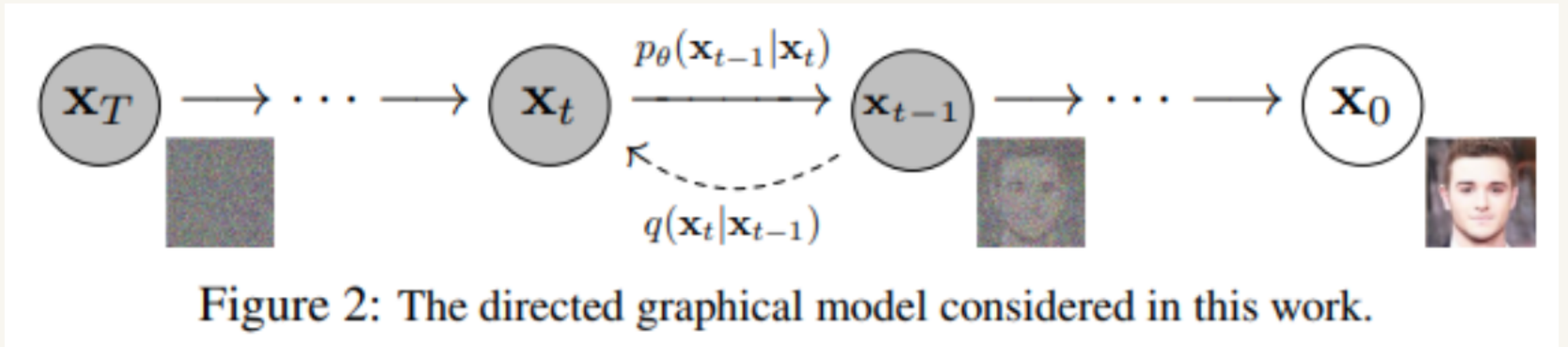
What changes during generation?

Training changes the model's weights.

Sampling changes the noisy image or its smaller representation.

A sharper picture can still have the wrong composition.

From noise to an image



DDPM forward and reverse processes. Read the arrow labels: generation proceeds from noise toward an image.

The model predicts. The sampler takes a step.

The model predicts from the current noisy image.

The sampler uses that prediction to calculate the next image.

The process repeats until generation finishes.

Different steps use different noise levels

Near pure noise, little image information remains.

Near the end, the picture is much easier to recognize.

The schedule chooses which noise levels to visit.

What can the model predict?

Some models predict the noise.

Others predict a clean image.

Others predict a direction of change, often called velocity.

The sampling method must match what the model was trained to predict.

Working with a smaller representation

512 × 512 × 3

Original image in Stable Diffusion v1

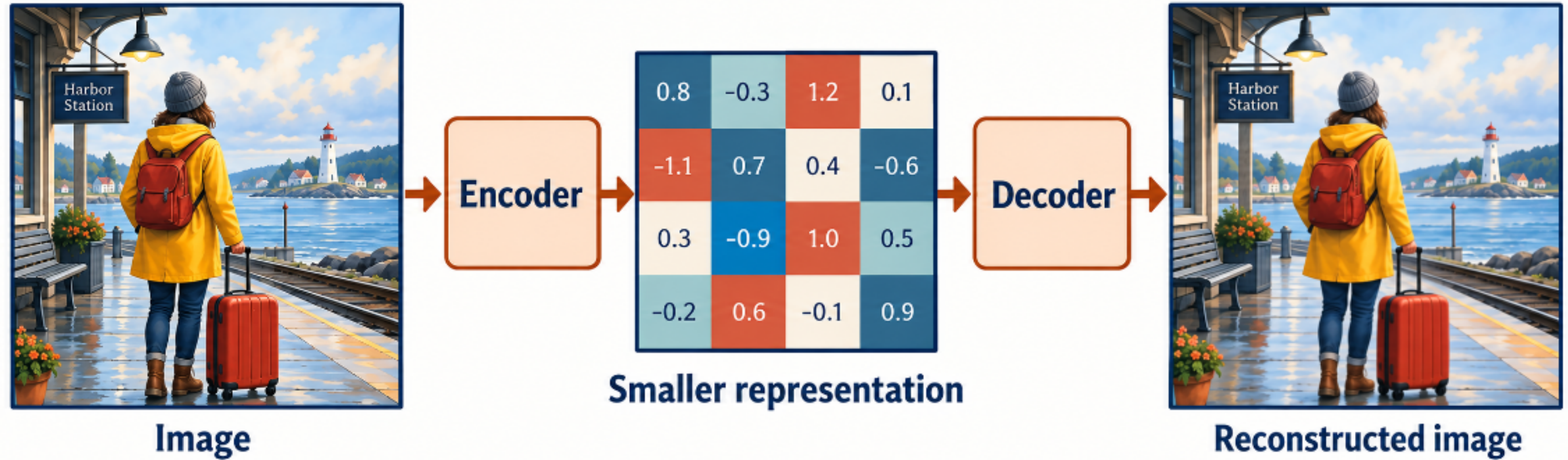
64 × 64 × 4

Smaller working representation in that model

A latent is a learned representation of an image.

It has fewer values for the denoising model to process.

How an image gets compressed



The encoder makes a smaller representation.

The decoder turns it back into pixels.

Conceptual diagram. The colored grid is not real latent data.

Rebuilding an image and making a new one

Reconstruction: start with an image and try to rebuild it.

Generation: start with a sampled state and make a new image.

A good reconstruction does not guarantee that random latents look good.

What does a VAE add?

A VAE is a variational autoencoder.

Its encoder describes a range of possible latent values.

Training balances image reconstruction with a constraint on those values.

This is one way to organize a space of compressed images.

Which details survive compression?

Fine lines and small details can disappear during compression.

Try encoding and decoding an image to see what survives.

Look closely at the suitcase handle and coat outline.

Edges and texture: paper, paint, and ink

A Cut paper



B Soft paint



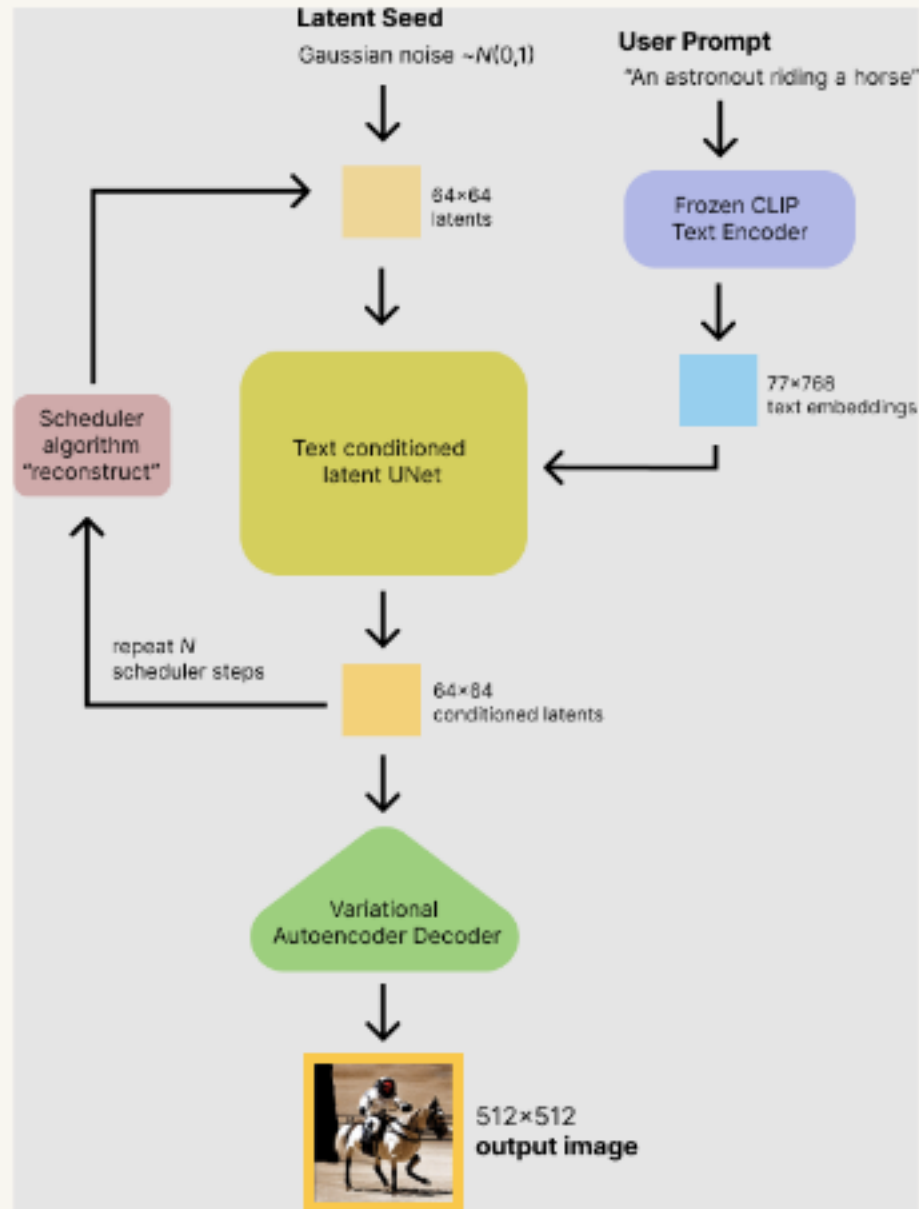
C Ink lines



Illustrations made for this class. The pictures are similar, but not identical.

Hard edges separate shapes; soft edges can merge them.

The parts of an image generator



The seed starts the random numbers.

The text encoder turns words into features.

The network predicts each update.

The sampler applies the update.

The decoder turns the result into pixels.

SD v1 pipeline diagram, Patrick von Platen / Hugging Face.

How a model uses words

The prompt is split into small pieces called tokens.

A text encoder turns those pieces into numbers.

The image model uses those numbers to help make the picture.

How CLIP links pictures and words

CLIP learns from paired pictures and captions.

It gives matching pairs similar numerical features.

Some image generators use its text encoder.

CLIP connects pictures and words. It does not draw the picture itself.

A similar-looking image can still be wrong

The picture may show a station and a traveler.

It may still put the suitcase on the wrong side.

Check specific details, not just the overall resemblance.

How attention shares information

Self-attention

Parts of the same input share information.

For example, one image area uses information from another.

Cross-attention

One input uses information from another input.

For example, image areas use information from prompt words.

An image area can use information from words



This coat area can use information about “yellow.”

Attention gives different weights to different features.

Illustrative connections, not a measured attention map.

Attention as a weighted mix

attention = weighted sum of values

Example weights: traveler 0.70, yellow 0.20, dusk 0.10

Multiply each value vector by its weight, then add them.

The weights add up to one.

The numbers are invented to show how weighted mixing works.

Why position matters

“Dog bites person” differs from “person bites dog.”

In an image, left and right are different places.

The model needs information about word order and image position.

Discussion 2: Does it match the instructions?

Goal: yellow coat, red suitcase on the left, station roof on the right.

90 sec: list the details you would check.

90 sec: could the picture look good and still get these details wrong?

Hue, lightness, and figure–background separation

Two different properties

Hue: yellow, red, blue.

Lightness: how light or dark a color appears.

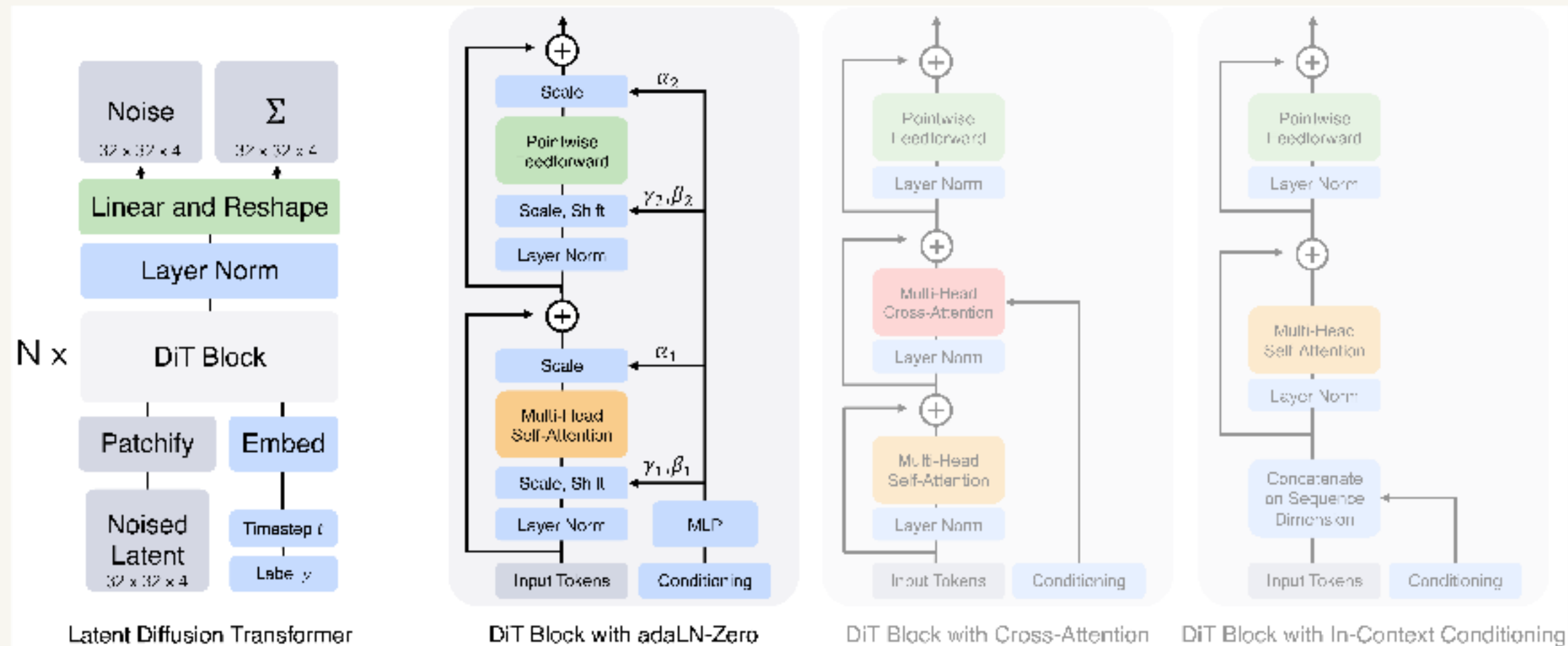
A coat against a wall

Different hues can have similar lightness.

Change the background value to separate the silhouette.

Try a grayscale check before adding texture.

A transformer can remove noise



DiT architecture, Peebles & Xie (2023), Figure 3. Original DiT uses class conditioning.

Smaller patches mean more work

The image representation is split into patches.

Smaller patches create more tokens.

Full self-attention compares every token with every other token.

Twice as many tokens means four times as many pairwise scores.

The network and the generation method

Autoregressive generation

Predict the next token.

Use the tokens already generated.

Diffusion generation

Predict an update to a noisy state.

Repeat at different noise levels.

Making an image one token at a time

Turn an image into a sequence of tokens.

Predict the next token from the earlier ones.

Turn the completed sequence back into pixels.

Flow matching: learning how to move

$$\mathbf{x}_t = (1 - t) \mathbf{z} + t \mathbf{x}$$

A simple training path joins noise $\mathbf{z}$ to an image $\mathbf{x}$.

The model learns the direction of change along that path.

At generation time, a solver follows the learned directions.

Following the model's direction

Ask the model which direction to move at the current time.

Take a small step in that direction.

Repeat to move from noise toward an image.

Example: next state = current state + step size × predicted direction

Where does randomness enter?

A run can start with random noise.

Some samplers also add random noise during the run.

Others follow a fixed path once the starting state is fixed.

Four terms that describe different parts

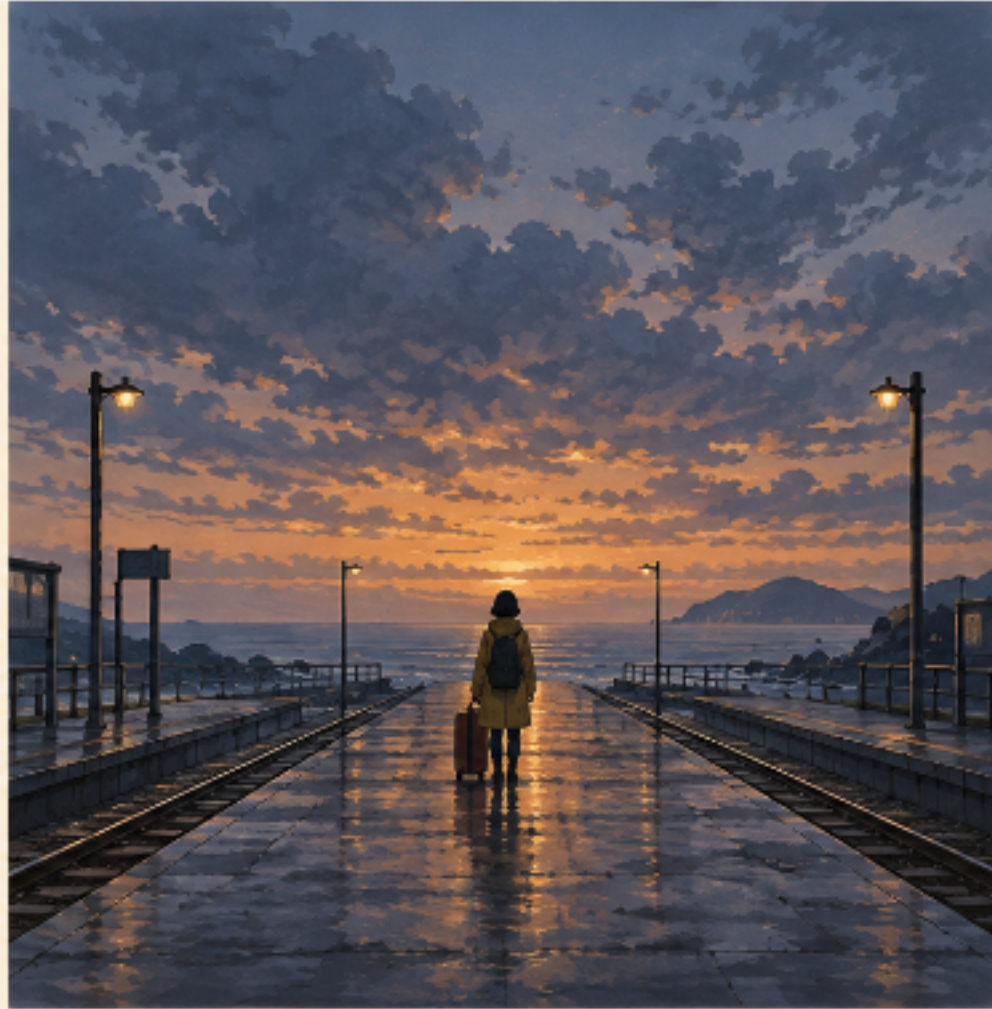
Term	What it describes
Transformer or U-Net	The network used to process information
Pixels or latents	The representation the network works on
Diffusion or flow matching	How we train the model to generate
Sampler or solver	How we calculate the next step

Different seeds can give different pictures

A Figure on the left



B Figure in the distance



C Figure on the right



Illustrative alternatives. These are not outputs from a measured seed comparison.

A seed starts the random numbers. It does not specify a camera position.

How to repeat a result

Save the model, prompt, references, and settings.

Reset the random generator before each comparison.

Use the same software and hardware when exact matching matters.

Character continuity: shape, costume, and props

Keep the same

Coat, hair, and suitcase design.

Which hand holds the suitcase.

The light and colors in the scene.

Change when needed

Camera angle and distance.

Pose and expression.

How much detail the shot needs.

A character sheet defines what must stay



Classroom illustration • Three views of one costume and prop

Record silhouette, coat length, buttons, hair, and carrying hand.

Guidance: how strongly to follow the prompt

Make a prediction with the prompt.

Make a prediction without the prompt.

Guidance increases the difference between them.

How guidance combines two predictions

$$\text{prediction} = u + s (c - u)$$

u: prediction without the prompt

c: prediction with the prompt

s: guidance scale in this formula

At $s = 1$, the result is c . Above 1, we push beyond c .

A small guidance calculation

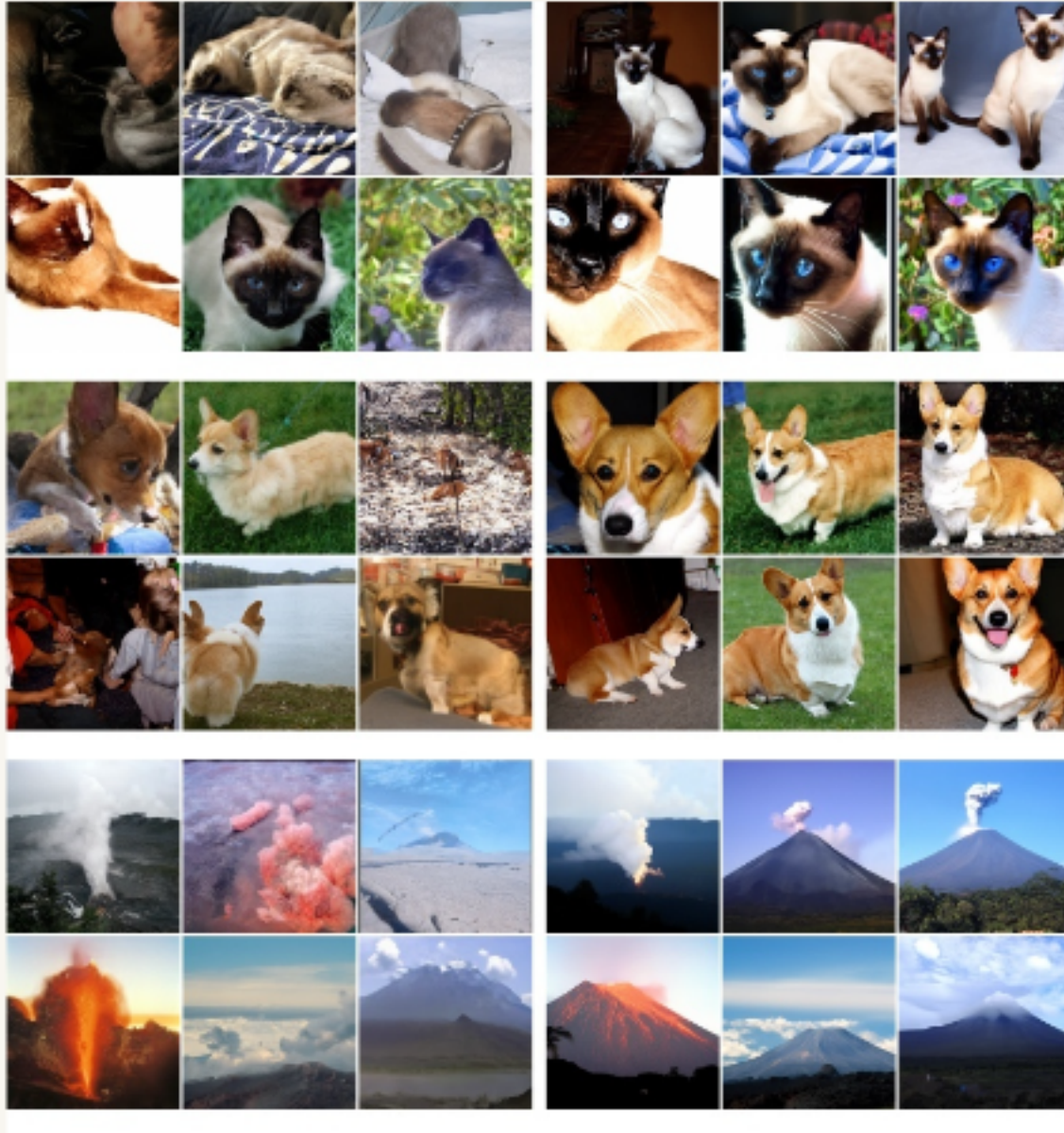
$$\text{guided prediction} = u + s(c - u)$$

Let $u = 2$ and $c = 3$ for one prediction coordinate.

$$\text{At } s = 1: 2 + 1 \times (3 - 2) = 3$$

$$\text{At } s = 4: 2 + 4 \times (3 - 2) = 6$$

What can go wrong with strong guidance?



The subject may become easier to recognize.

Colors may become too strong.

Different results may start to look alike.

Ho & Salimans, Figure 3: unguided samples at left, guided samples at right.

How a negative prompt can help

$$\text{prediction} = n + s (c - n)$$

c: prediction with the positive prompt

n: prediction with the negative prompt

Some systems push away from n and toward c.

A negative prompt cannot guarantee that an object disappears.

Three guidance settings

Scale 2.5



Scale 7.5



Scale 10.5



Hugging Face documentation examples. Seed control is not documented for this displayed comparison.

Discussion 3: Clear subject, wrong mood

The coat is yellow, but the picture feels harsh and crowded.

90 sec: how would you compare guidance settings fairly?

90 sec: how would you judge the colors and mood?

What should we compare?

Keep the same model, prompt, and other settings.

Use the same set of seeds at each guidance value.

Compare prompt accuracy, appearance, and variety.

15 steps and 50 steps

15 inference steps



50 inference steps



Published SD v1-4 pair: seed 1024, guidance 7.5, same prompt. Hugging Face (2022).

Which details changed? Did they improve the picture?

Steps and sampler: two different settings

Step count

How many updates to take.

Usually affects how long generation takes.

Sampler and schedule

How to calculate each update.

Which noise levels to visit.

A progress picture or a finished result?

A progress image comes from the middle of one run.

A step comparison uses finished images from separate runs.

Changing the step count can also change the schedule.

Generation cost: count model evaluations

20 steps × 2 predictions = 40 evaluations

Classic CFG uses conditional and unconditional predictions.

Batching can compute the pair together.

Latency also depends on image size, model, and hardware.

This arithmetic is a simple CFG example, not a timing benchmark.

Some models need only a few steps

1–4 steps

SD-Turbo's documented operating range

Guidance disabled

SD-Turbo's standard text-to-image setup

Distillation changes the trained model.

Settings from an older base model may no longer apply.

Changing the model



Different training images can lead to different results.

The network and training method also matter.

The same settings may behave differently in another model.

Selected class-conditioned DiT samples, Peebles & Xie (2023), Figure 1.

LoRA: change fewer model parameters

$$\Delta W = B A$$

Full $1,000 \times 1,000$ update: 1,000,000 values.

Rank 8: $(1,000 \times 8) + (8 \times 1,000) = 16,000$.

The small learned update is added to the base weight.

Example dimensions • Rank controls the size of the update.

Reference input or learned subject adaptation?

Method	Where the change happens
Text / image prompt	Conditioning during generation
IP-Adapter	Image features enter separate attention
LoRA	A low-rank update changes selected weights
DreamBooth	Subject examples are used for finetuning

Conditioning and weight updates are different forms of control.

ControlNet: edges, depth, and pose

Control input	What it helps specify
Edges	Building outline and object boundaries
Depth	Near/far arrangement of surfaces
Pose	Locations of body joints
Text prompt	Appearance, subject, and other requested details

Spatial controls can specify structure more directly than style words.

Starting with a drawing

A Starting drawing



B Similar layout



C Larger changes



Illustrations of editing choices, not measured outputs at specific strength settings.

What must stay the same: the pose, the layout, or only the subject?

Editing strength: preservation and change

Less added noise often preserves more of the starting drawing.

More noise usually allows larger changes.

The prompt and model still affect what survives.

Local editing starts with a mask

$$\text{Final} = \text{mask} \times \text{edit} + (1 - \text{mask}) \times \text{original}$$

Mask = 1: use the edited region.

Mask = 0: keep the original region.

Soft edges blend the transition.

A repair to the suitcase should not require a new face.

A layered image-editing workflow

Set composition with a sketch or spatial reference.

Repair a local region with a mask.

Finish color and lettering in editable layers.

Save the generated base, edited regions, and final composite.

A video needs to stay consistent



Top: the coat and suitcase stay the same.

Bottom: the coat and bag change between shots.

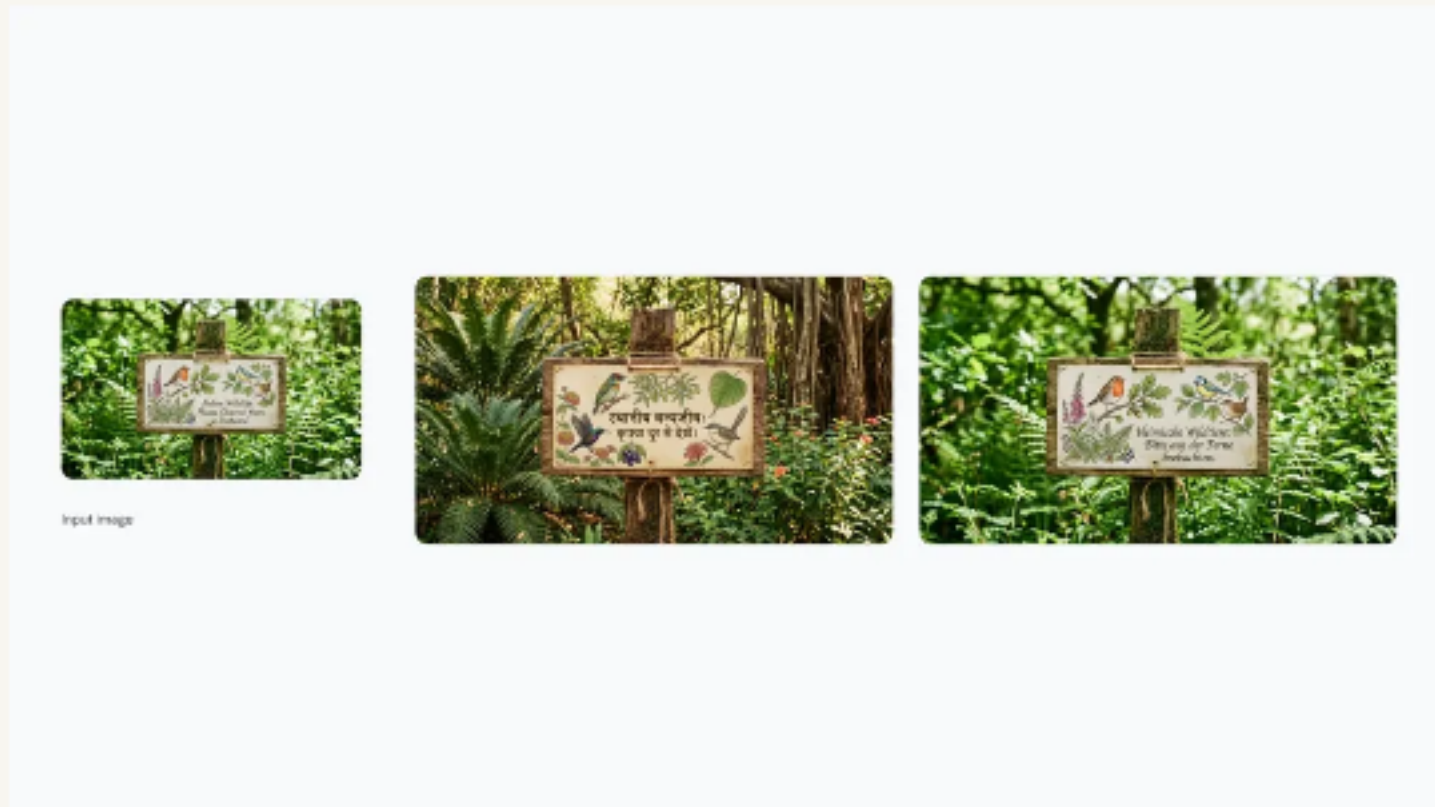
Illustrated examples. These are not frames from a tested video model.

2026: Text inside pictures

Nano Banana 2 can place and translate words in an image.

Design use: try a poster in another language.

Check every word and the spacing.

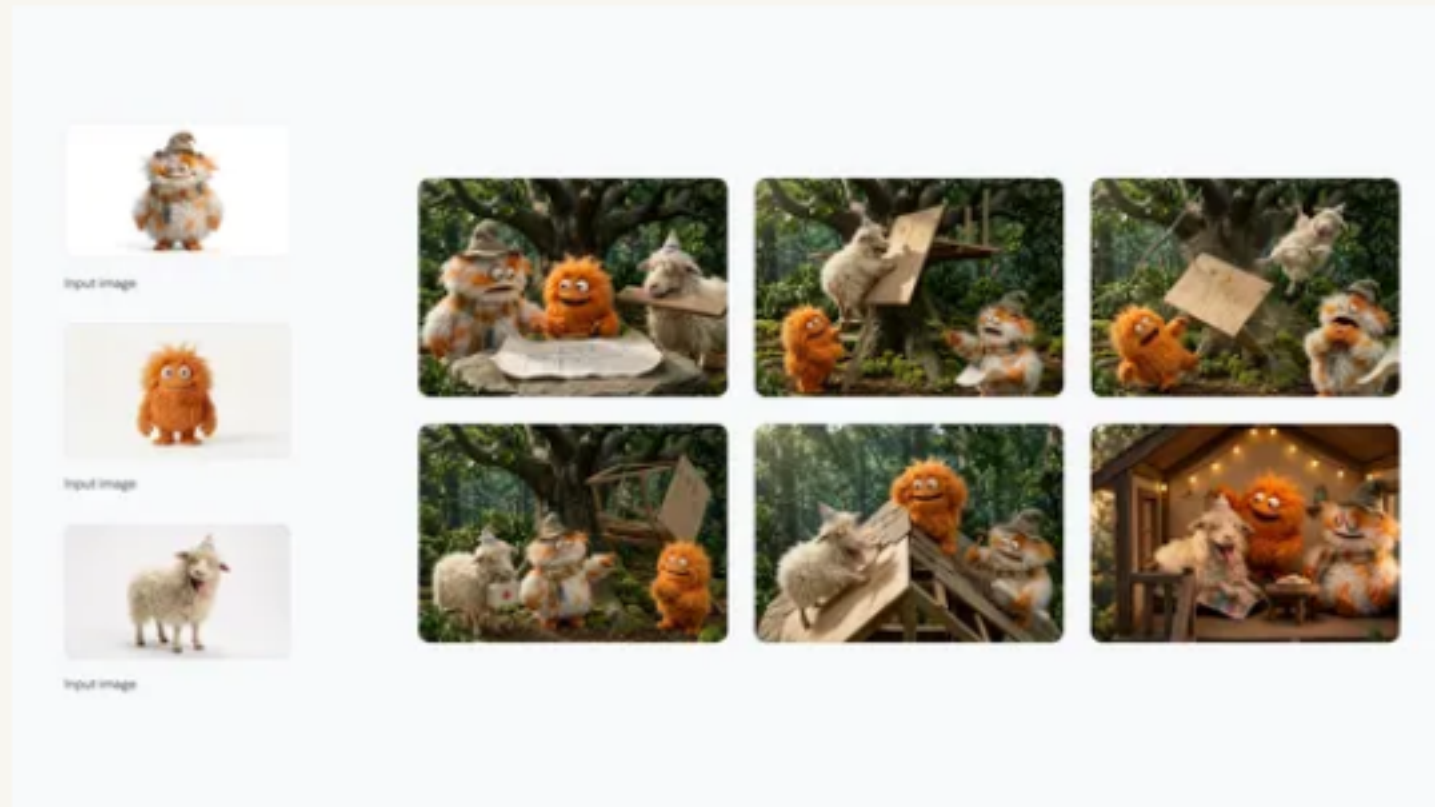


2026: Keeping characters recognizable

Nano Banana 2 supports character consistency across images.

Design use: build a short storyboard.

Check faces, clothes, and props in every shot.



September 2026: GPT Image 2.5

Sunburst

OpenAI's most capable image model.

Image generation and editing.

Flare

Designed for faster everyday image work.

Image generation and editing.

8 September 2026 snapshots • Compare the same task and references.

2026: Give the video model examples

What changed

Seedance 2.0 accepts text, pictures, sound, and video as inputs.

It can generate short video with sound.

For our station scene

Use a character picture and a camera-movement clip.

Check whether the movement fits the story.

A reference can show a movement that is hard to describe in words.

2026: Edit a video by describing a change

What changed

Gemini Omni Flash supports video editing with words.

Examples include changing lighting or replacing an object.

For our station scene

“Keep the traveler. Make the station lights warmer.”

Check what changed in the background too.

Review the whole clip after an edit, including its sound.

Video review: frame, motion, sound

Pass	A concrete test
Individual frames	Count objects and inspect faces and lettering
Motion across frames	Track the suitcase; check walls during camera motion
Sound with picture	Compare footsteps, door motion, and sound onset
At the cut	Check screen direction and prop position

A convincing frame is only one part of a convincing sequence.

2026: Video, sound, and keyframe control

FLUX 3 Video: up to 20 seconds with native audio.

Keyframes can specify important moments.

Check motion, dialogue timing, and continuity.

Initial video release: 4 August 2026 • BFL developer report

GPT-6 Astra: Help build an interactive scene

What it can help with

Works with text and image inputs.

Can write code and use tools across several steps.

Try it for our station scene

Build a scene that responds to a visitor.

Review the result, then ask for a specific change.

Describe → build with tools → inspect → revise

Astra: turn a design instruction into a test

Describe the behavior

“Click the suitcase to fade the lights in two seconds.”

“Keep the traveler and the camera fixed.”

Check the built result

Does one click start one fade?

Do repeated clicks leave the scene in a clear state?

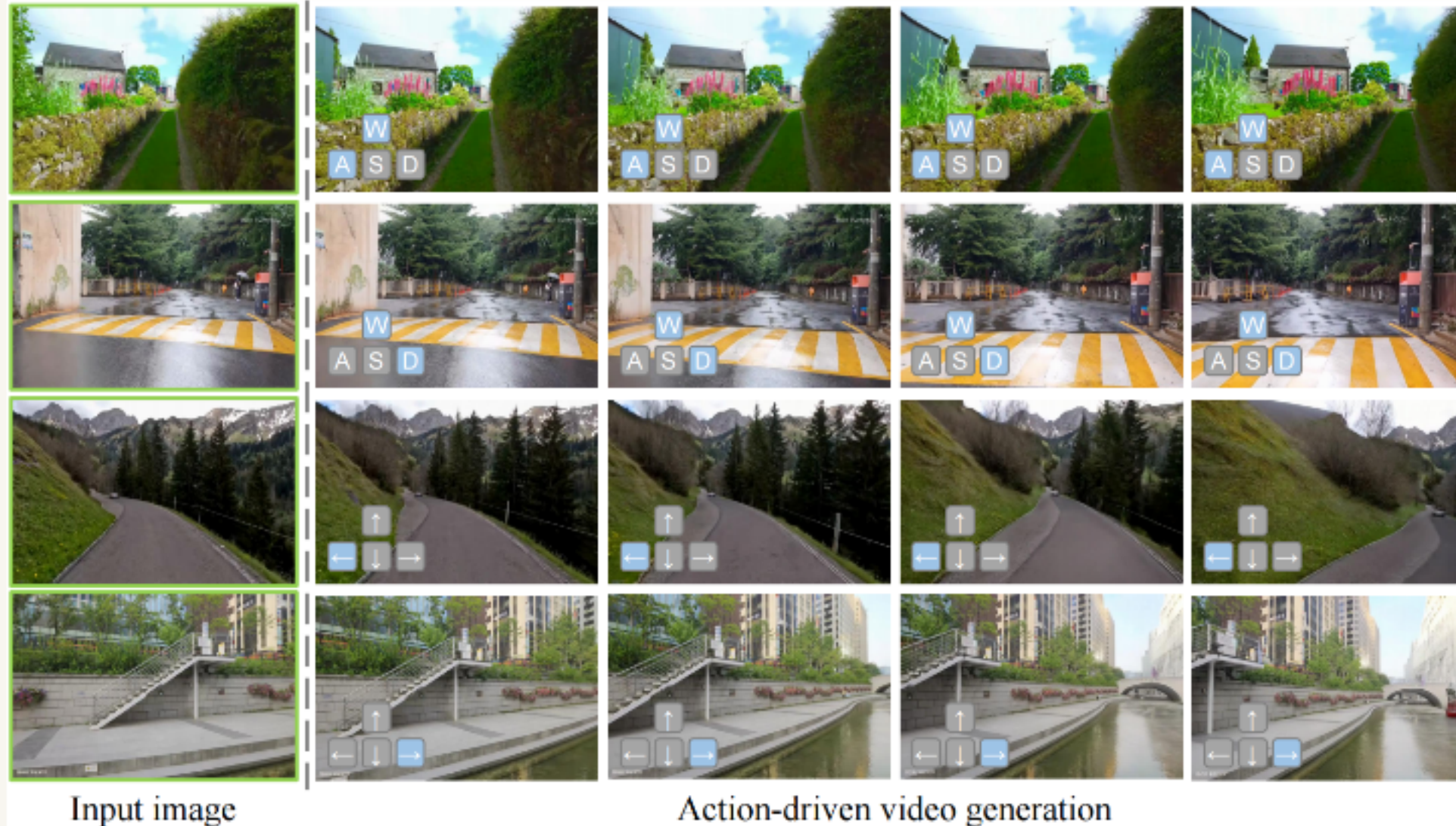
Use code and tools to build, run, inspect, and revise the interaction.

World models: what is being predicted?

Representation	Example output
State	Next position or a learned numerical state
Pixels	Next video frames conditioned on actions
3D scene	A scene representation that can be rendered
Action condition	The proposed movement affects the prediction

The representation determines which questions the model can answer.

Astra: A picture that responds to movement



Give it a starting image and movement controls.

It predicts the next part of the video.

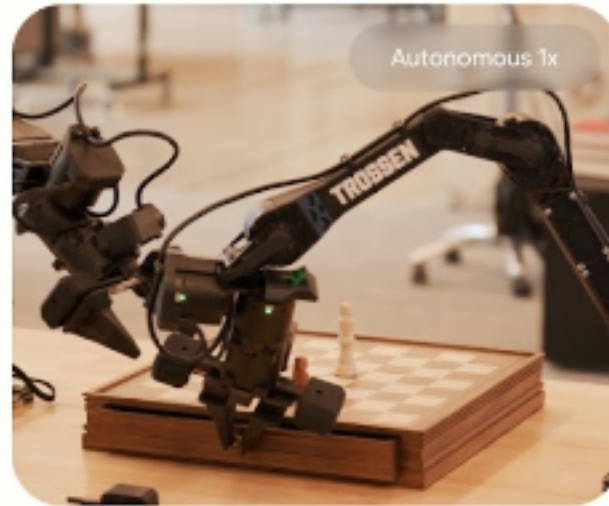
Astra authors' examples • ICLR 2026 • Preprint: December 2025

World models: test a round trip

Action	What should remain consistent?
Enter a doorway	Opening size and wall position
Turn to look away	Objects outside the view still matter
Turn back	The doorway returns in the same place
Leave the room	The route agrees with the starting scene

A plausible next frame can still lead to an inconsistent place.

VLA: See the scene, read the task, act



Vision + language → robot actions.

Gemini Robotics 2 adds whole-body control.

Robot actions need a coordinate system

What the camera sees

The prop appears 40 pixels left of center.

The image alone does not specify a distance in meters.

What the robot must do

Move relative to a defined robot or world frame.

Observe again after the movement.

“Move left” needs a viewpoint, a distance, and a stopping condition.

Predicting an action's result and taking action

System	In an interactive exhibition
Assistant	Helps write code for an interactive scene
World model	Predicts what a proposed movement might look like
VLA	Produces commands to move a real object
Combined system	May compare predictions, act, then observe again

A VLA does not have to contain a separate world model.

Discussion: Design an experience that responds

Imagine a visitor moves the red suitcase in our station scene.

Should the system explain, predict a new scene, or move a real prop?

What should happen if it misunderstands the visitor?

60 seconds • Choose one response and explain how the visitor controls it.

Which setting might help?

What looks wrong?	What could you try?
The traveler is too large	Ask for a wider view or give a layout reference
The look changes between images	Use the same style reference and color instructions
The colors look too strong	Compare lower guidance values, if supported
The character keeps changing	Use character references and inspect each result
One small area looks wrong	Try a local edit and check the surrounding area

How to make a fair comparison

Part of the test	Example
Question	Does guidance change variety and subject clarity?
Keep fixed	Model, prompt, size, sampler, steps
Change	Guidance only
Repeat	Use the same set of seeds at each value
Judge	Subject clarity, appearance, and variety

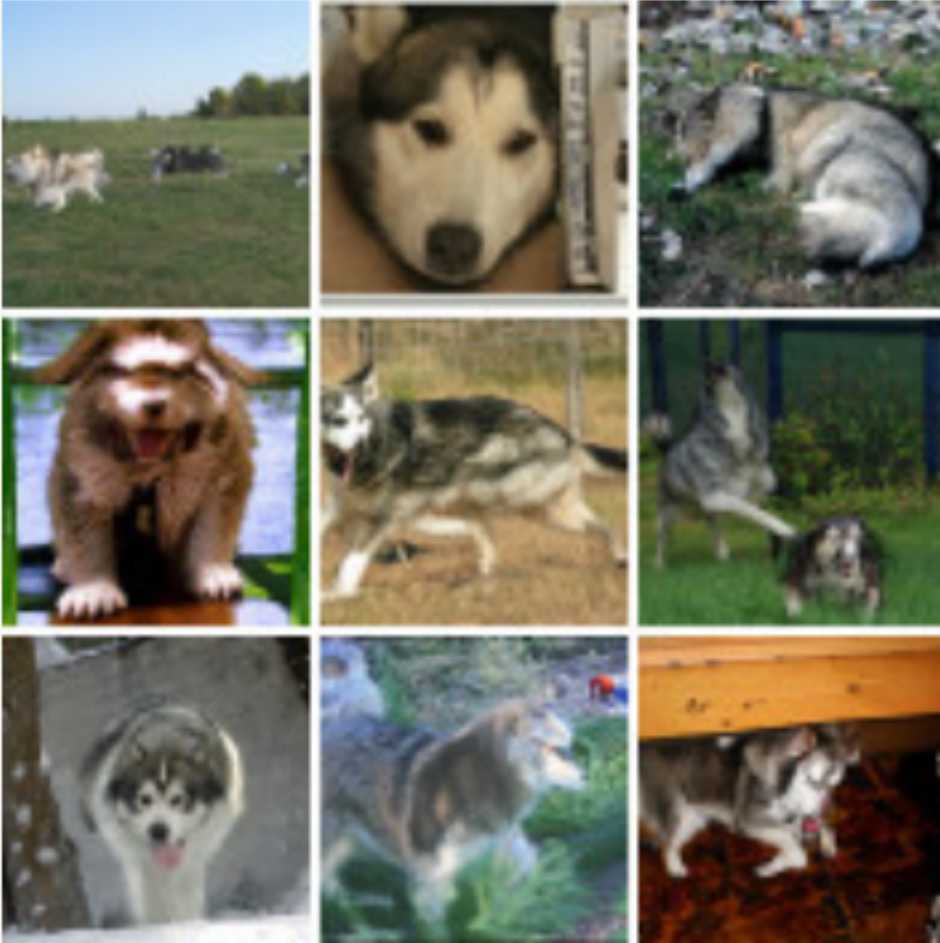
Content, form, and variation

Question	Concrete evidence
Content	Correct count, objects, colors, and relations
Form	Silhouette, lightness, edges, and focal point
Variation	Different usable views across the matched seeds
Purpose	A clear poster and a dream sequence may differ

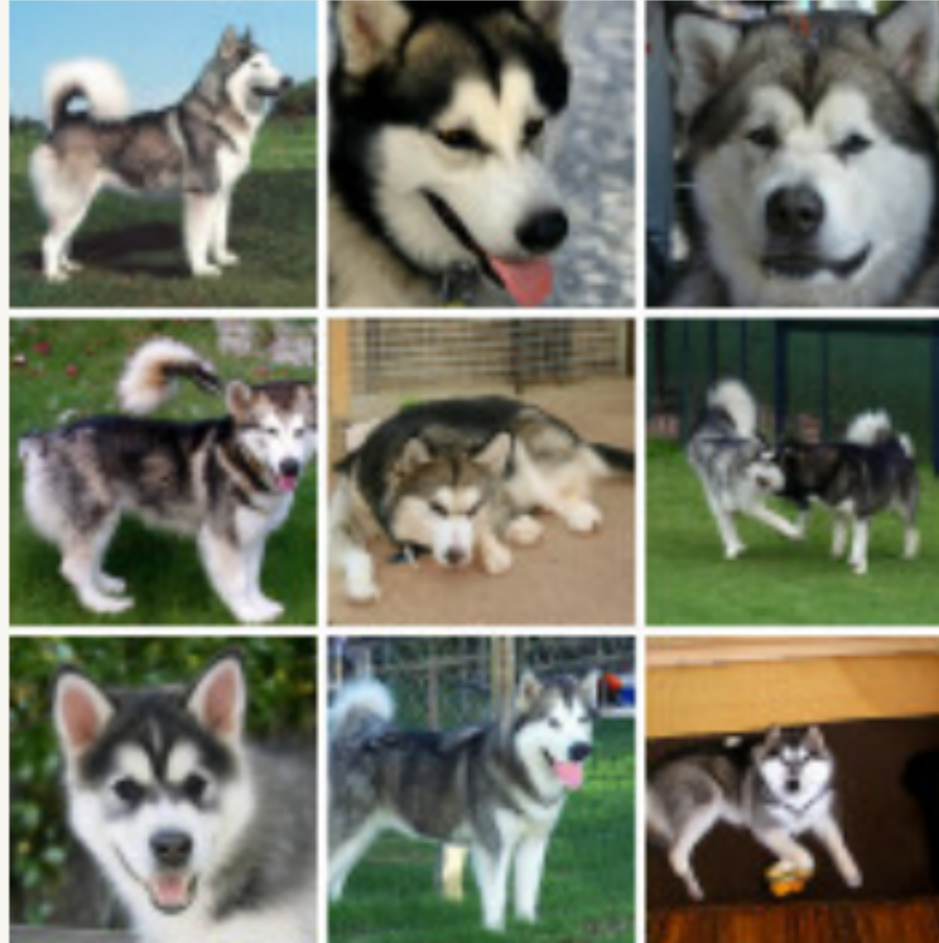
Describe one visible fact before giving a preference.

Discussion 4: Compare the pictures

No extra guidance: $w = 0$



Guidance: $w = 1$



Guidance: $w = 3$



Matched details from Ho & Salimans, Figure 6. ImageNet 64×64 , malamute class, shared random seed.

Which pictures would you choose for a clear introduction? For a strange dream?

Reading a guidance grid

Compare across a row

Same seed: follow changes in the face and outline.

Mark colors or edges that become exaggerated.

Compare within a column

Different seeds: compare pose and composition.

Record repeated patterns and usable variation.

A row isolates settings; a column shows variation across seeds.

Discussion 5: Choose a look for our scene

30 sec: choose a camera view and drawing style for our scene.

60 sec: explain your choice using two details in the pictures.

30 sec: name one change you would try next.

Three questions before we finish

Why can the same prompt give different pictures?

Why might stronger guidance make the picture worse?

What would you save so you can repeat a result?

Suggested answers

Different random starting states can lead to different pictures.

Strong guidance can reduce variety or create harsh colors and other problems.

Save the model, prompt, references, seed, and generation settings.

A method for the next image

Choose the structure.
Control the change.
Check the result.

Prompt and references describe the target.

Seed, guidance, and sampler shape the output.

Art decisions determine what makes the image useful.

University lectures behind this lesson

CMU 10-423 / 623 / 723

Generative AI · Spring 2026

Diffusion, text-to-image, adaptation and world models

MIT 6.S184

Flow and Diffusion Models · 2026

Sampling, flow matching, guidance and latent spaces

Stanford CS231n

Generative Models · Lecture 14, 2025

Generative modeling and diffusion

Stanford CS148 / CS178

Color, cameras and composition

Established art and imaging foundations

Papers, demonstrations, and further reading

Generation and editing

DDPM · LDM · CFG · LoRA · SDEdit · ControlNet

Sources are linked in the notes of the relevant slides

2026 image and video

Nano Banana 2 · Seedance 2.0 · Omni Flash · FLUX 3

Developer demonstrations and classroom design examples

Assistants and world models

OpenAI GPT-6 Astra · Astra world-model paper

Two separate systems with different tasks

Robotics

Vision-language-action · Gemini Robotics 2

Perception and instructions mapped to robot actions