

A decade of denoising turned noise into worlds

Diffusion models, 2015 → 2026 · image & video

Yikai Wang

▶ 90' full talk

▶ 45' core cut

t = 0.00 step 50/50

Background: LDM sample · Rombach et al., 2021

From a slow image generator to the generative engine of foundation models



It works

DDPM · 256^2 faces, ~1000 steps



Text → image on a gaming GPU

Latent / Stable Diffusion



Minute-long video

Sora · spacetime-patch DiT

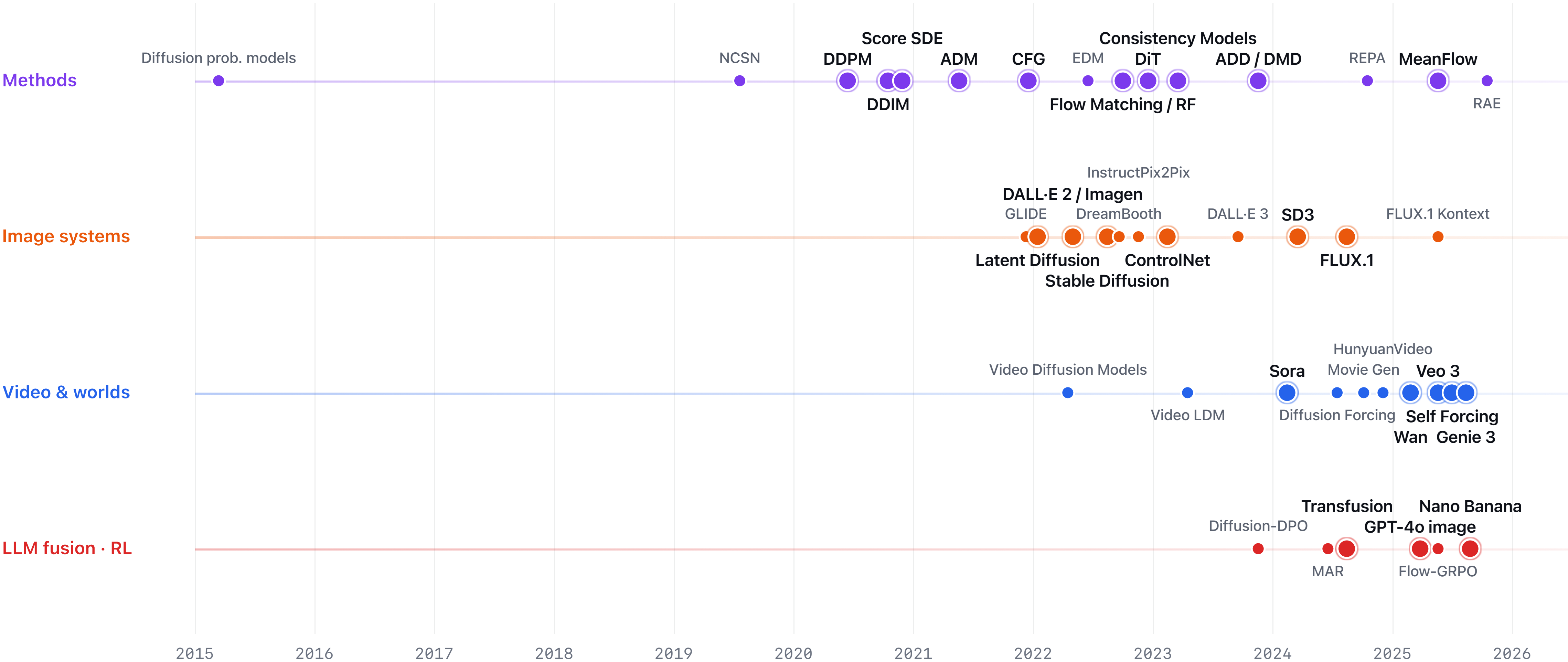


Real-time interactive worlds

Genie 3 · [architecture undisclosed](#)

→ One recipe all along: learn to remove noise at every noise level, then generate by denoising pure noise.

From 1000 steps to **real-time worlds**



→ Five quiet years, then everything at once: most milestones land between 2020 and 2025.

A short primer, nine chapters, one trend summary

0

Primer

noise → denoise ·
score · ODE · CFG

1

Origins

2015 · NCSN · DDPM ·
DDIM · Score SDE

2

Beating GANs

ADM · CFG · EDM

3

Text-to-image

GLIDE · DALL·E 2 ·
Imagen · LDM/SD ·
DALL·E 3

4

Control

DreamBooth · LoRA ·
ControlNet ·
IP2P → Kontext

5

Architectures & objectives

DiT · Flow Matching ·
SD3 · FLUX · REPA → RAE

6

1000 steps → 1

solvers · Consistency ·
ADD/DMD · MeanFlow

7

Video I

VDM · Video LDM ·
Sora · Movie Gen ·
Hunyuan

8

Video II · worlds

Wan · Self Forcing ·
Veo 3 · Genie 3

9

LLM fusion · RL

DPO · GRPO ·
Transfusion · GPT-4o ·
Nano Banana

— PRIMER

Diffusion in **one picture**: learn to denoise, then denoise noise

forward noising

ϵ -prediction

score

SDE vs ODE

guidance

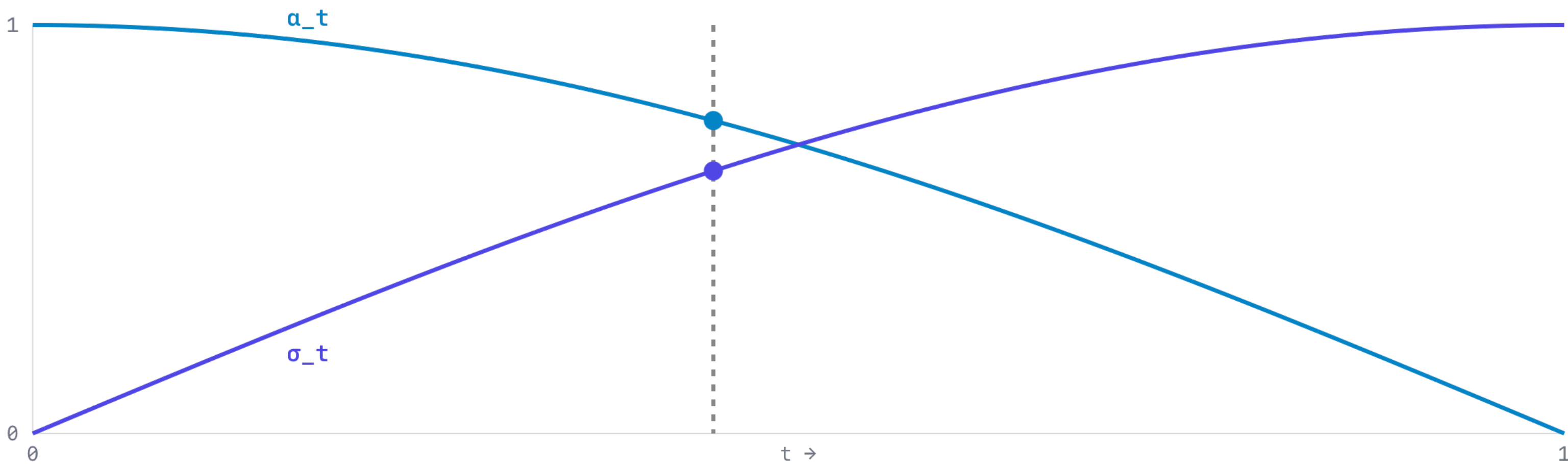
ϵ / x_0 / v

conditioning

design choices

Background: DiT-XL/2 samples, ImageNet 512² · Peebles & Xie, 2022 · appendix grids (mosaic; denoising is illustrative)

Destroy images with noise: $x_t = \alpha_t x_0 + \sigma_t \epsilon$



α_t (signal) = 0.77
 σ_t (noise) = 0.64
 $t = 0.44$ $\text{SNR} = \alpha^2 / \sigma^2 = 1.4$
variance-preserving: $\alpha^2 + \sigma^2 = 1$

t

0.44

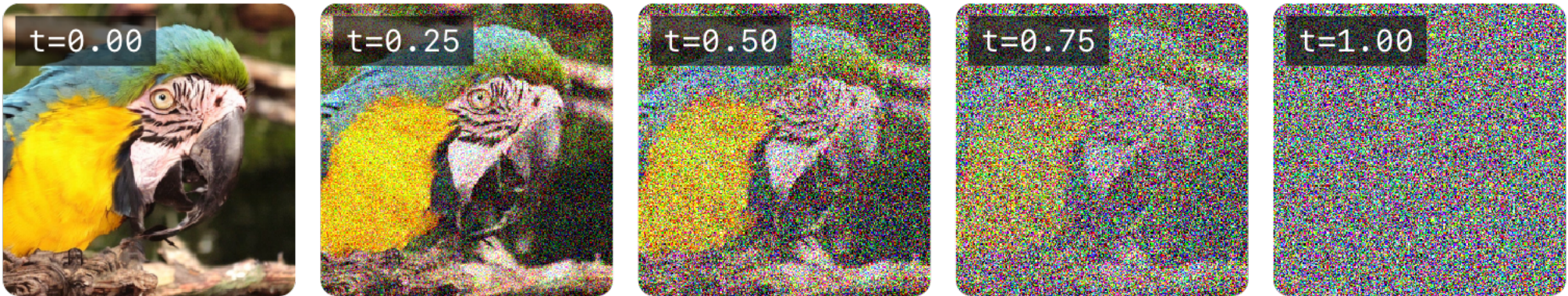
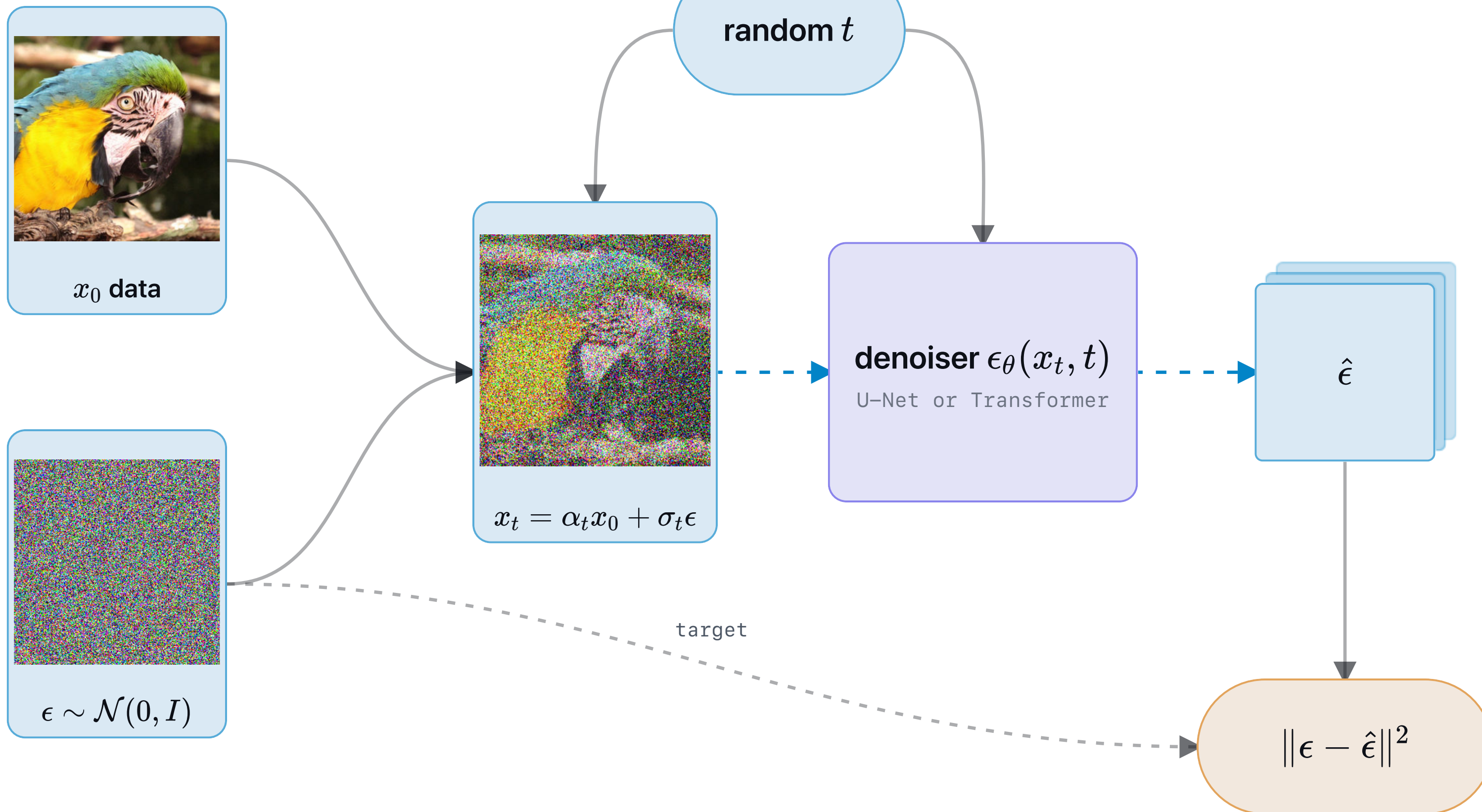


Image: DiT sample · Peebles & Xie, 2022
noise added live

➔ Any noise level t can be sampled in one shot, so training never has to simulate a chain.

Training is plain **regression**: predict the noise

Image: DiT sample · Peebles & Xie, 2022



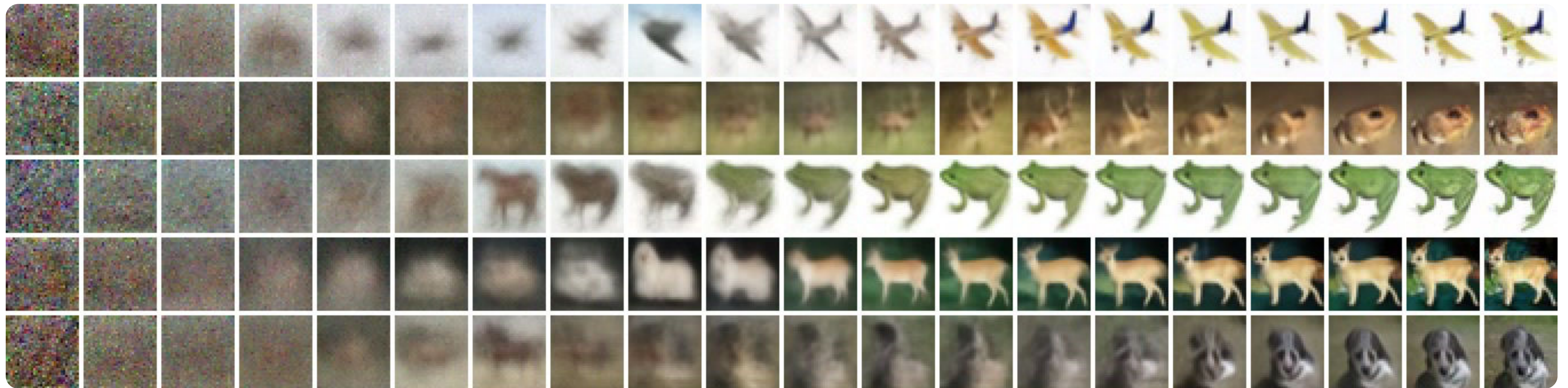
DDPM OBJECTIVE

$$\mathcal{L} = \mathbb{E}_{x_0, t, \epsilon} \|\epsilon - \epsilon_\theta(x_t, t)\|^2$$

- **No discriminator**, no adversarial game
- A **denoising autoencoder** trained at all noise levels at once

→ One MSE loss means it scales like supervised learning: bigger model + more data = better samples.

Generation = start from **pure noise**, denoise step by step



Ho et al., 2020 · Fig. 6 (rows cropped)

1000

NETWORK CALLS PER SAMPLE (DDPM)

noise → layout

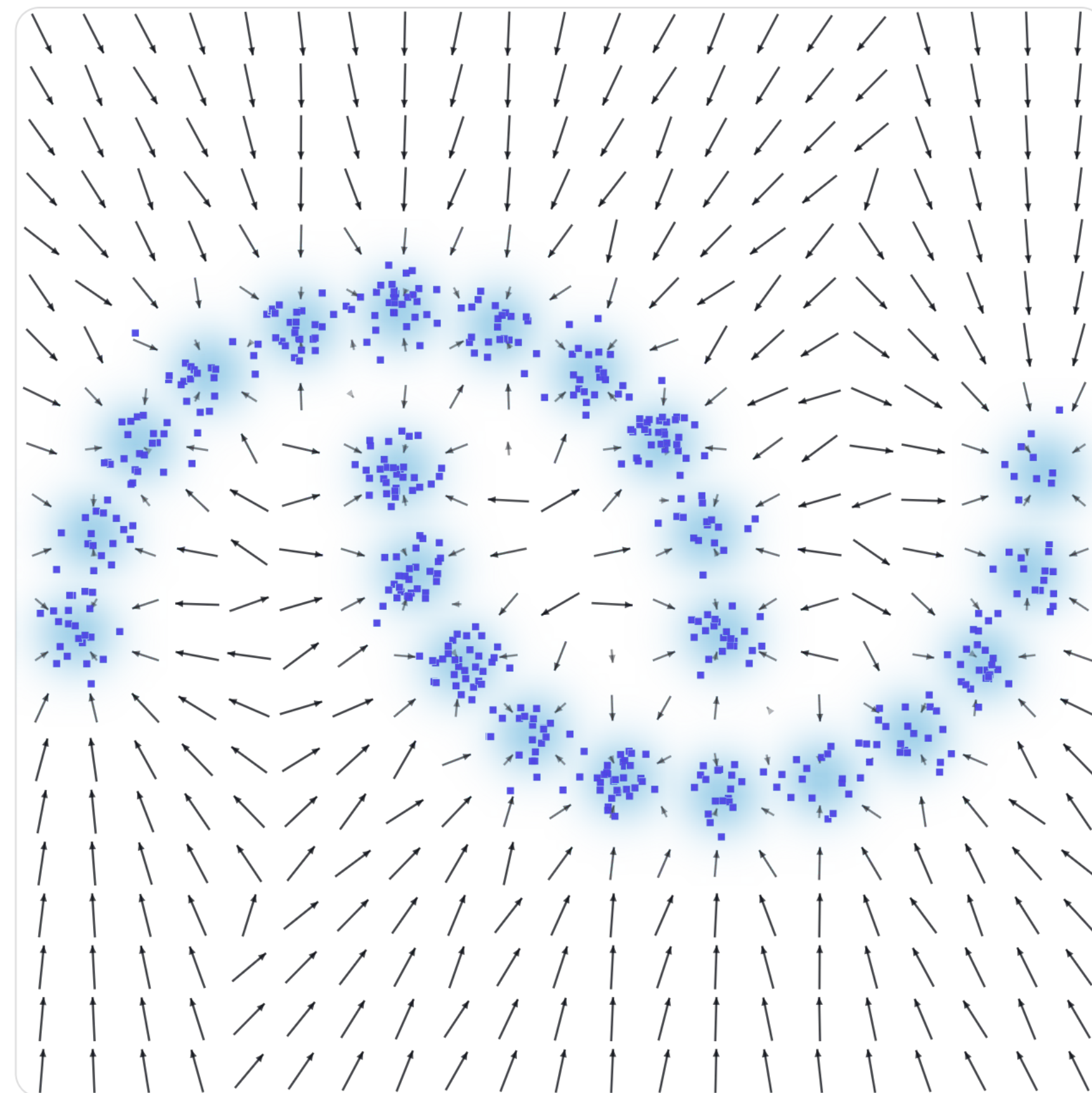
EARLY STEPS: GLOBAL STRUCTURE, COLOUR

→ detail

LATE STEPS: TEXTURE, EDGES

→ Like sculpting: start from a block of noise and chip away what does not look like an image.

A denoiser secretly knows **which way is data**



$t = 0.03$

arrows: $\sigma \cdot \nabla \log p_t(x)$
dots: Langevin samples
(noise annealed $1 \rightarrow 0$)

noise level t 0.03

DENOISING = SCORE ESTIMATION

$$\epsilon_{\theta}(x_t, t) \approx -\sigma_t \nabla_x \log p_t(x_t)$$

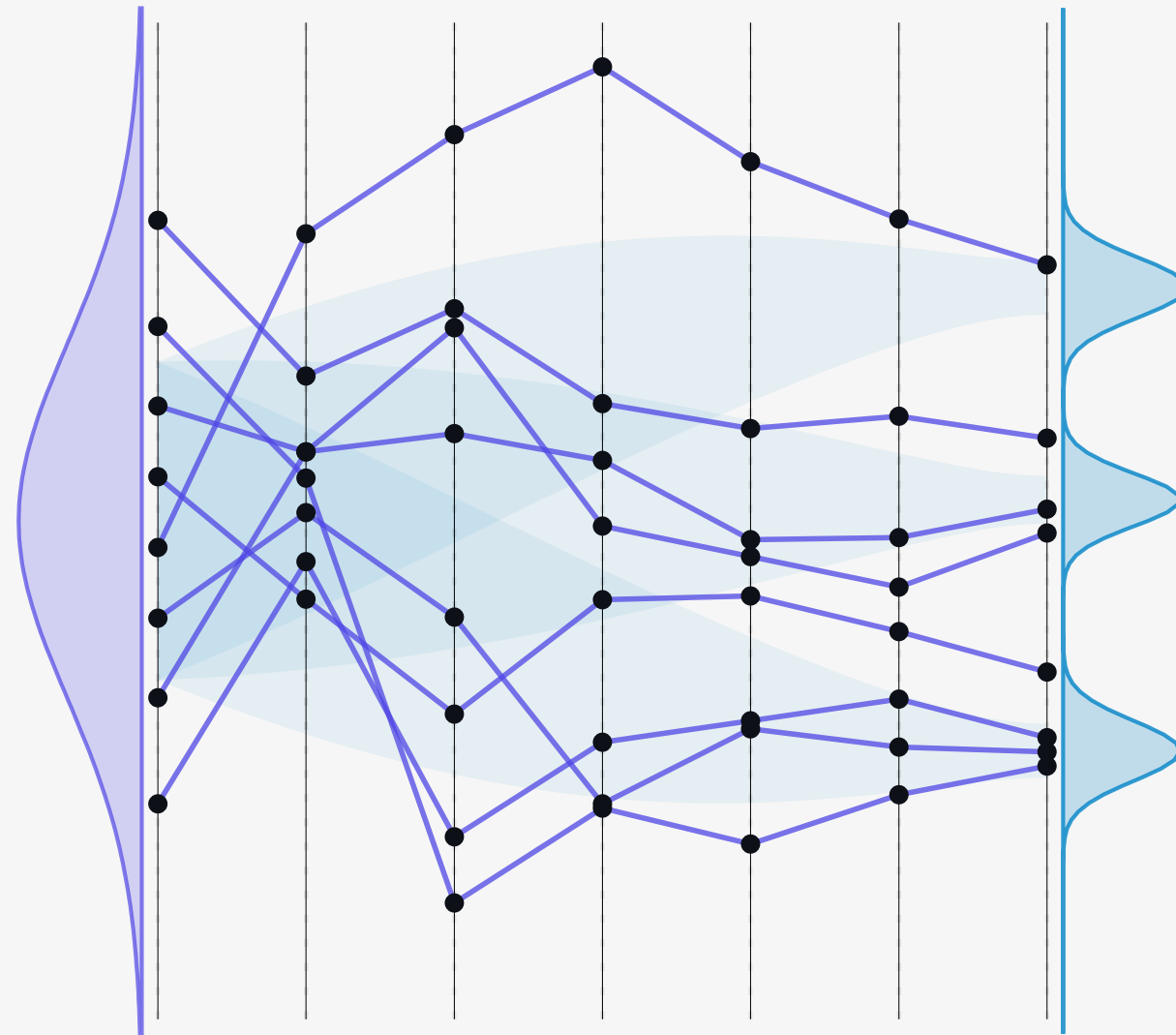
Arrows: the exact score of a toy 2-D mixture.
Dots: Langevin dynamics, i.e. follow the arrows plus a little noise, while the noise level anneals from $1 \rightarrow 0$.

→ Predicting the noise = knowing, at every noise level, the direction toward more likely data.

Papers switch between **three views**

2015 · DDPM 2020

Variational

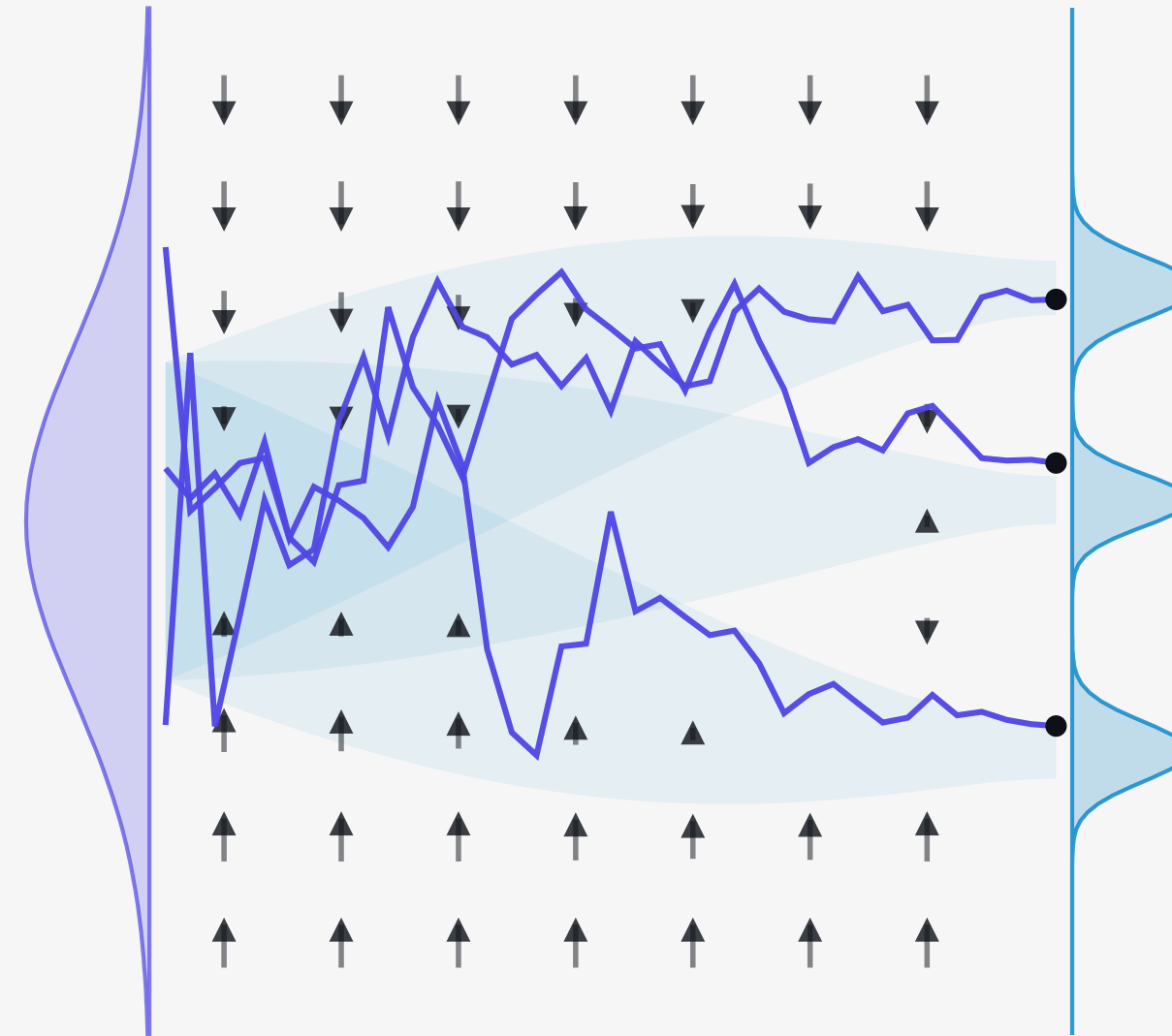


$$p_{\theta}(x_{t-1} \mid x_t)$$

learn each reverse step; walk the Markov chain back

NCSN 2019 · Score SDE 2020

Score

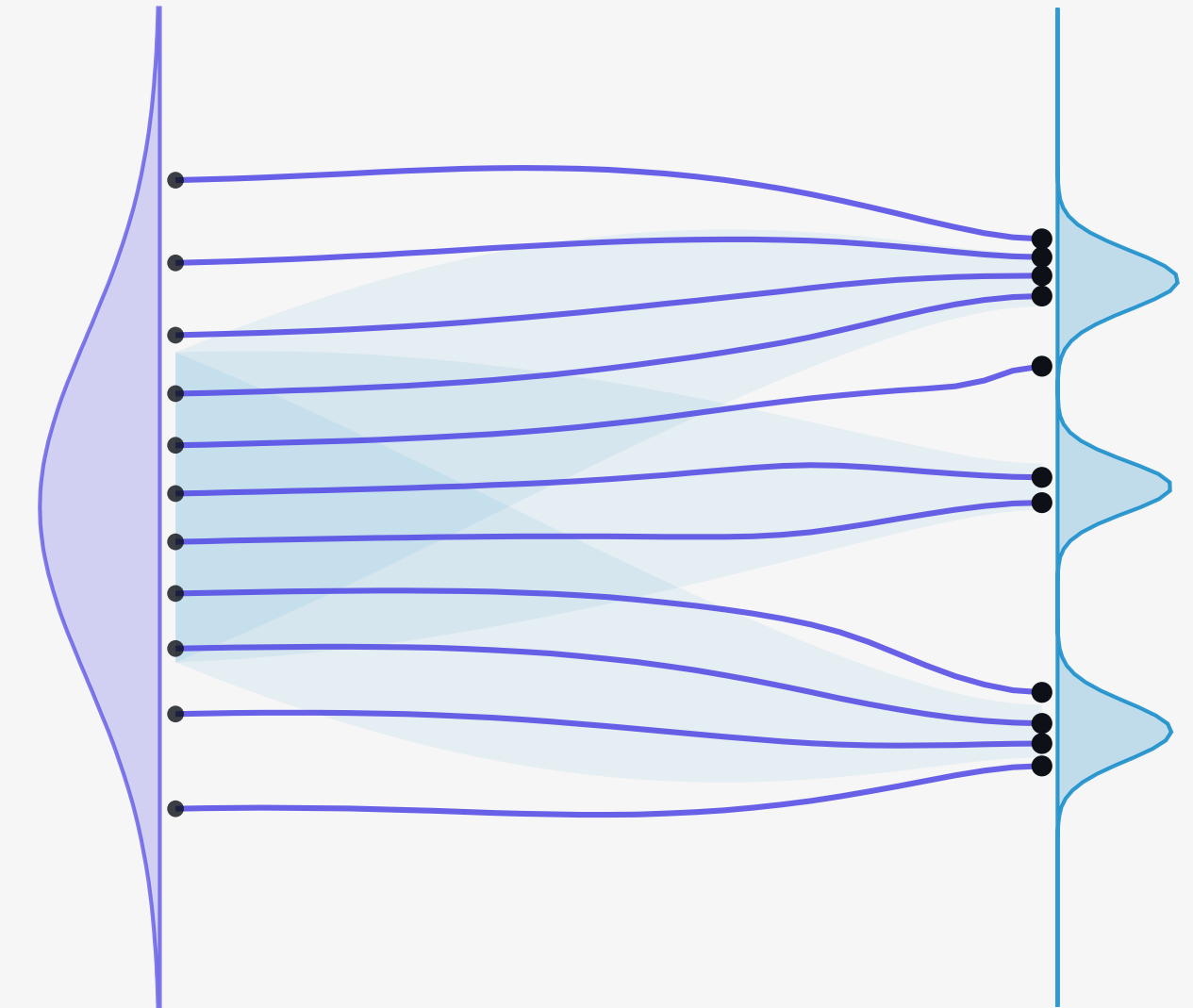


$$\nabla_x \log p_t(x)$$

learn the uphill direction; Langevin / reverse SDE

DDIM · PF-ODE · FM / RF 2022

ODE / flow

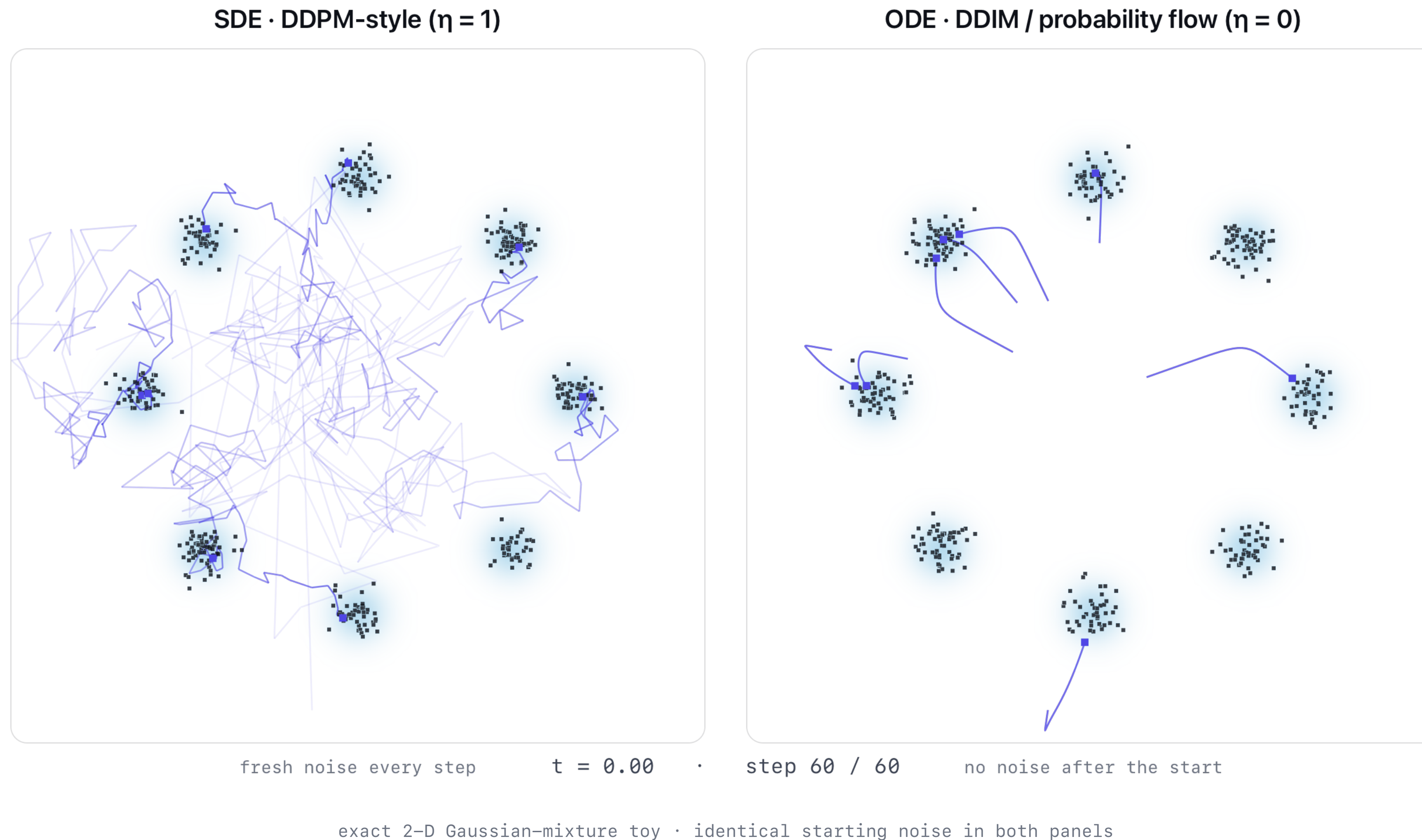


$$\frac{dx}{dt} = v_{\theta}(x, t)$$

learn a velocity field; integrate an ODE

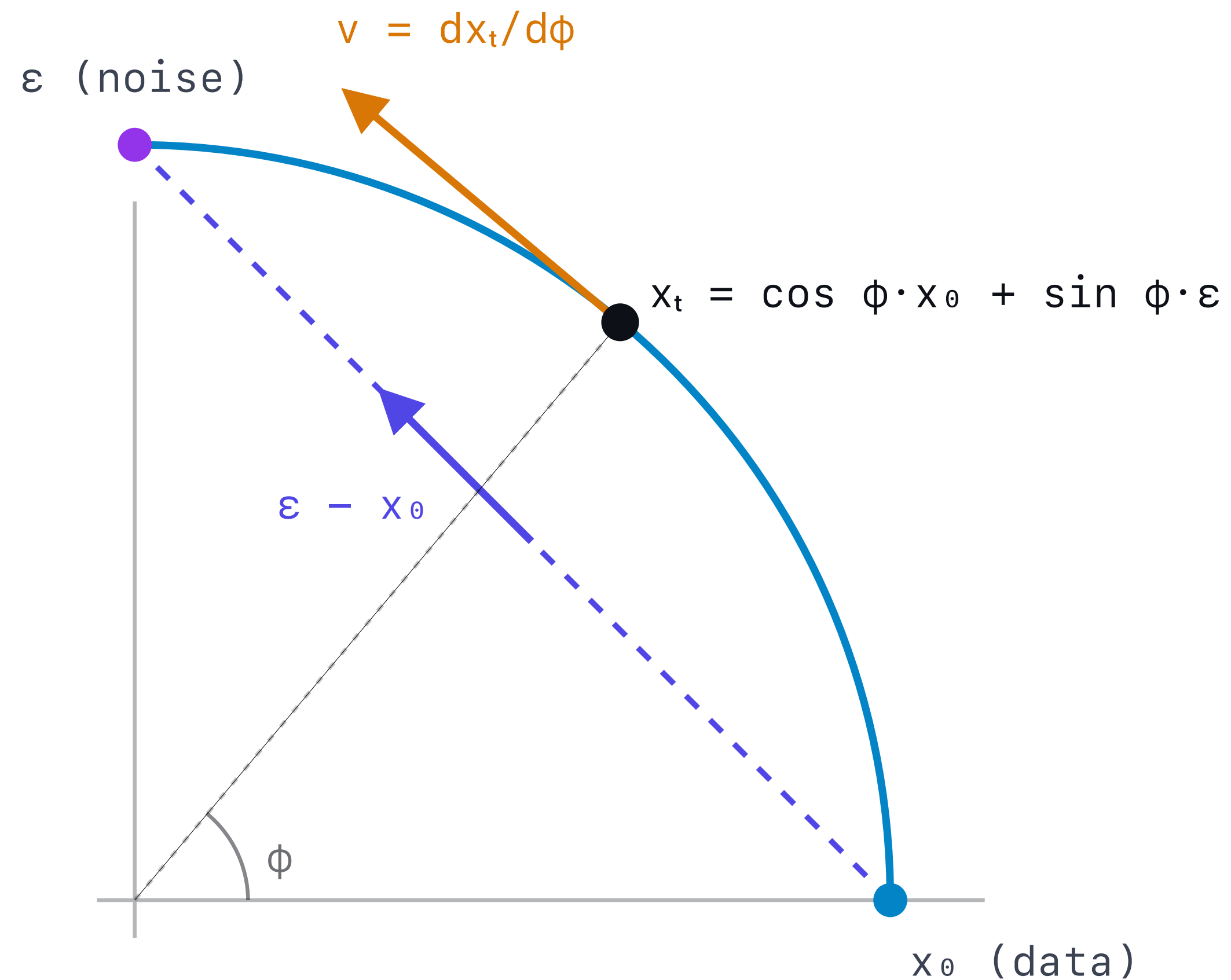
→ Same trained network, three readings, so results transfer freely between the three literatures.

SDE and ODE samplers reach the **same distribution**



→ The ODE has no randomness: sampling becomes numerical integration, and noise $\leftrightarrow$ image is invertible.

ϵ , x_0 or v : **same model**, different weighting



$\epsilon \rightarrow x_0$ (GIVEN x_t)

$$\hat{x}_0 = (x_t - \sigma_t \hat{\epsilon}) / \alpha_t$$

V-PREDICTION · TANGENT OF THE ARC

$$v = \alpha_t \epsilon - \sigma_t x_0$$

FLOW MATCHING · ALONG THE CHORD

$$x_t = (1 - t) x_0 + t \epsilon, \quad \frac{dx_t}{dt} = \epsilon - x_0$$

→ Knowing x_t , each target is a linear function of the others; they differ in error amplification per noise level.

Classifier-free guidance: **overshoot** the condition



w = 3

guided
 $p(x|c)^w p(x)^{(1-w)}$

■ class c (target)

■ other class

sample spread 1.01

guidance w

CFG AT EVERY STEP

$$\tilde{\epsilon} = \epsilon_{\theta}(x_t, \emptyset) + w [\epsilon_{\theta}(x_t, c) - \epsilon_{\theta}(x_t, \emptyset)]$$

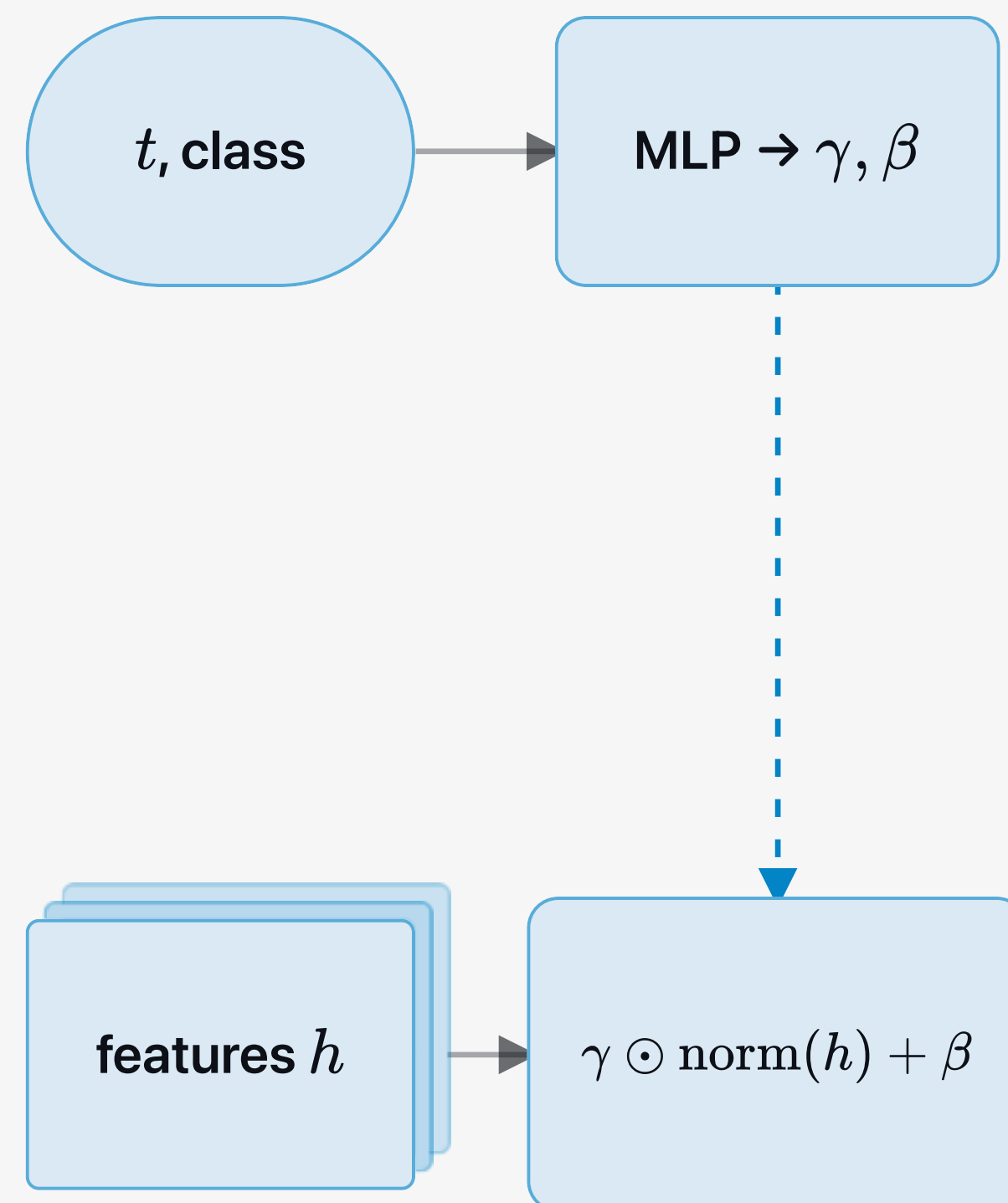
Train with the condition randomly dropped, so one network gives both terms. Click w: 0 = unconditional, 1 = plain conditional, > 1 = guided.

→ One setting trades diversity for fidelity to the prompt; almost every text-to-image or video system uses it.

Three ways a condition gets in

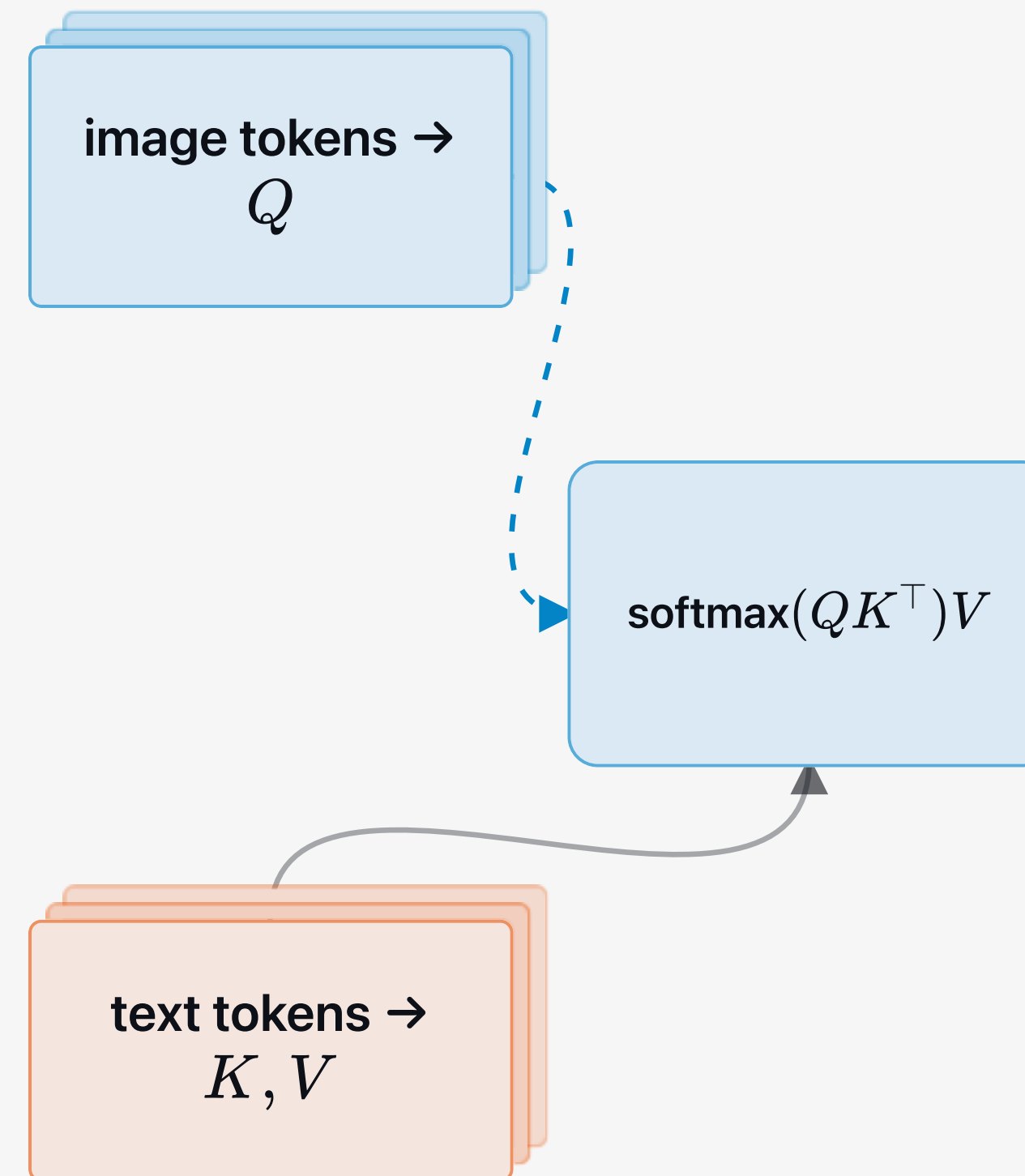
Modulate · AdaGN / adaLN

global vectors: timestep, class, pooled text



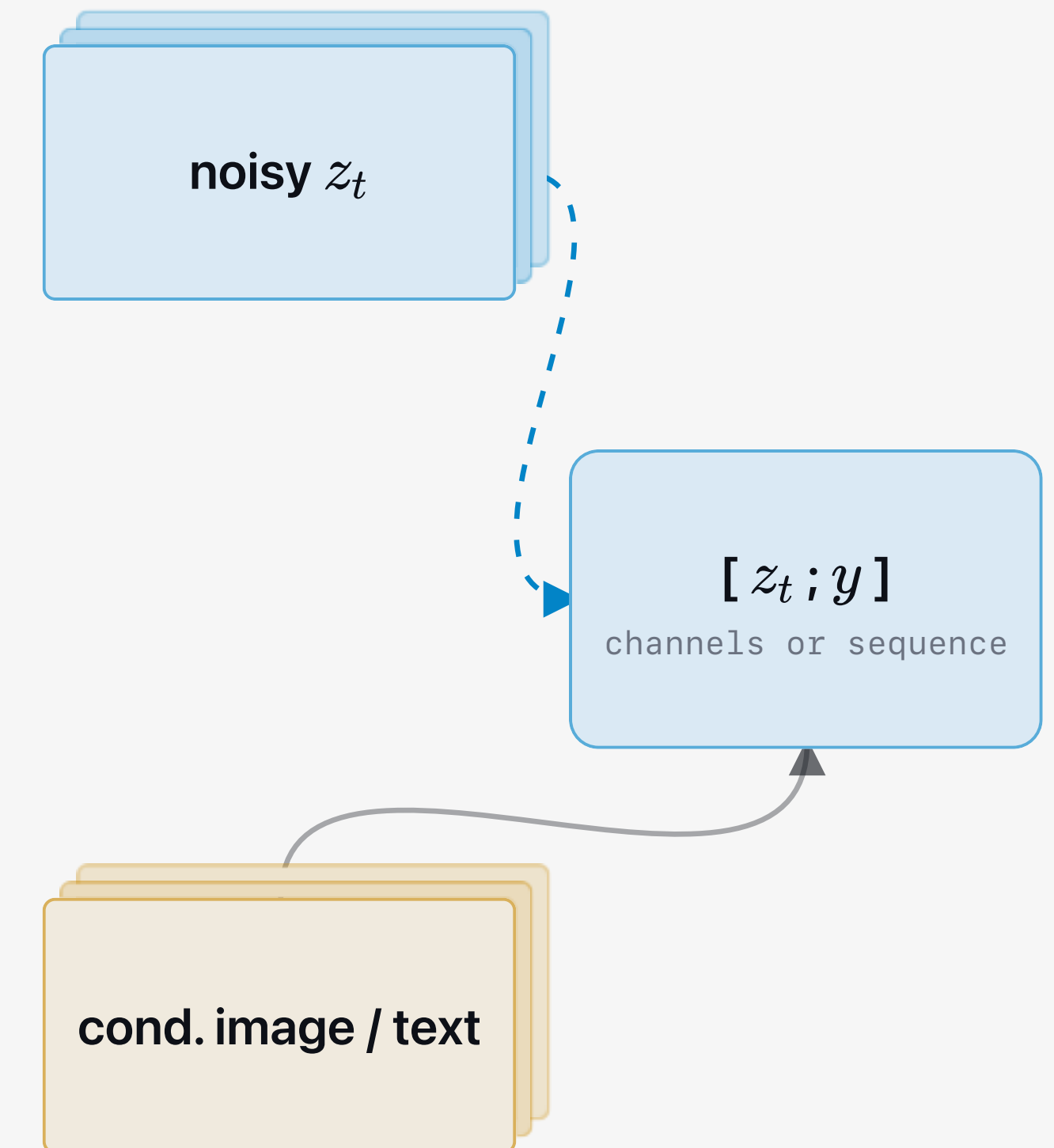
Cross-attention

token sequences: text from CLIP / T5



Concat / joint attention

spatial images, or text + image jointly



→ Global signals modulate normalization; sequences enter by attention; spatial signals are concatenated.

Stable, complete, editable, **but slow**

MSE

STABLE TRAINING

no generator–discriminator
balance, so scaling is safe

ELBO

COVERS THE DISTRIBUTION

likelihood-based: dropping modes
is penalized

∇t

EDITABLE

guide, constrain, invert, inpaint at
any step

$\times 1000$

THE PRICE

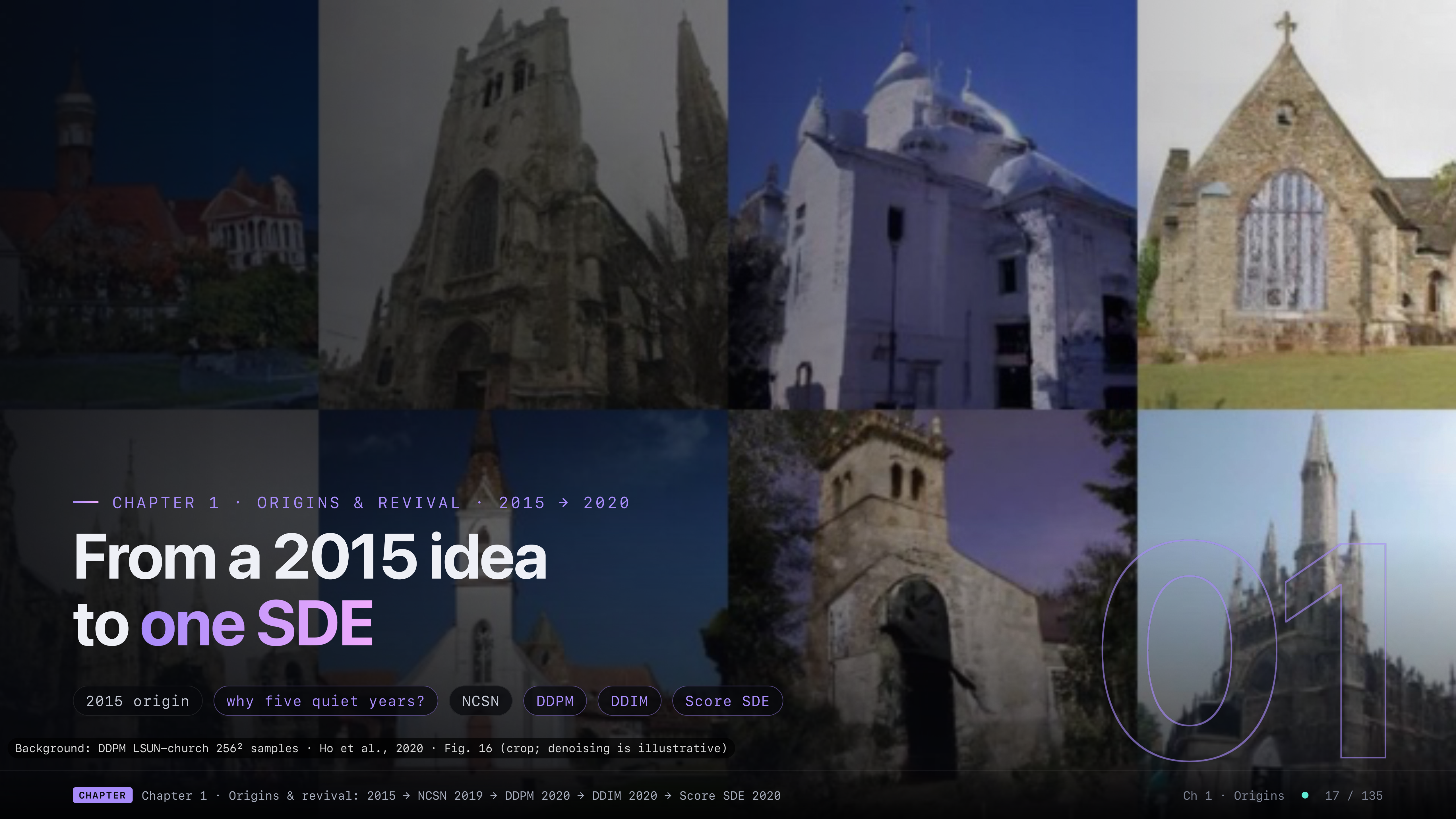
many network calls per sample →
chapter 6

→ The first three made it win; the fourth set the research agenda for six years.

Every milestone changes one of these choices

Objective	ε-pred → v-pred → EDM precondition. → flow velocity → x-pred returns	ch 1 · 2 · 5 · 6
Noise schedule	linear → cosine → EDM log-normal → logit-normal · res. shift	ch 1 · 2 · 5
Steps	1000 → DDIM 50 → ODE solvers 10–20 → distilled 1–4 → 1-step from scratch	ch 1 · 6
Guidance	classifier → classifier-free → distilled guidance	ch 2 · 3 · 6
Backbone	U-Net → DiT → MMDiT	ch 2 · 5 · 7
Where to denoise	pixels + cascade → VAE latent → 3D causal video VAE → semantic latent / pixels again	ch 3 · 5 · 7
Text encoder	class → CLIP → T5 → (M)LLM → diffusion inside the LLM	ch 3 · 5 · 9
Time (video)	factorized U-Net → spacetime DiT → causal, streaming	ch 7 · 8

➔ Read every milestone in this talk as a move along one of these rows.



— CHAPTER 1 · ORIGINS & REVIVAL · 2015 → 2020

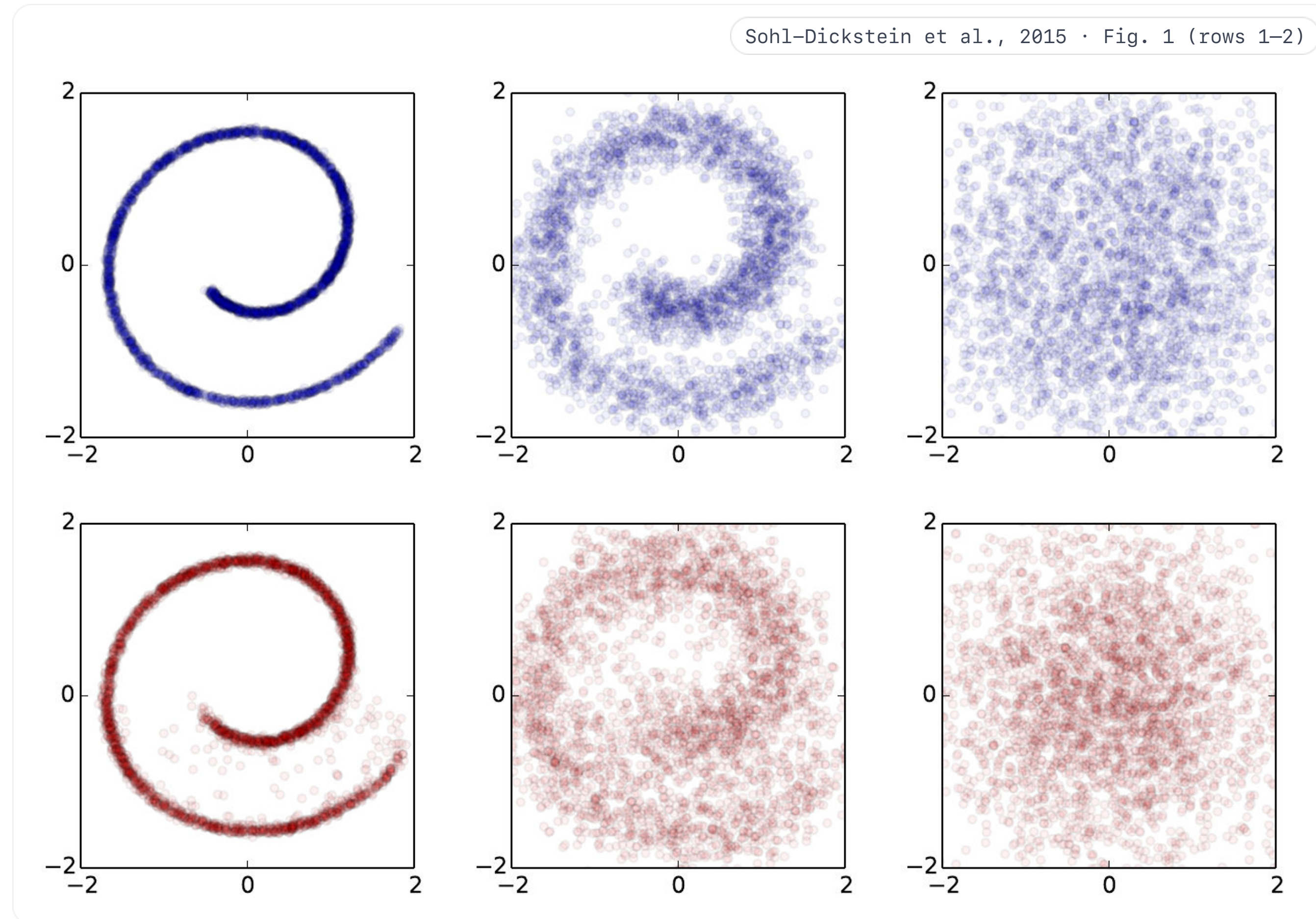
From a 2015 idea to one SDE

- 2015 origin
- why five quiet years?
- NCSN
- DDPM
- DDIM
- Score SDE

Background: DDPM LSUN-church 256² samples · Ho et al., 2020 · Fig. 16 (crop; denoising is illustrative)

01

2015: learn to **undo the noise**



$t = 0 \cdot T/2 \cdot T$ · **Top** forward $\rightarrow$ · **Bottom** learned reverse $\leftarrow$

FORWARD STEP · FIXED

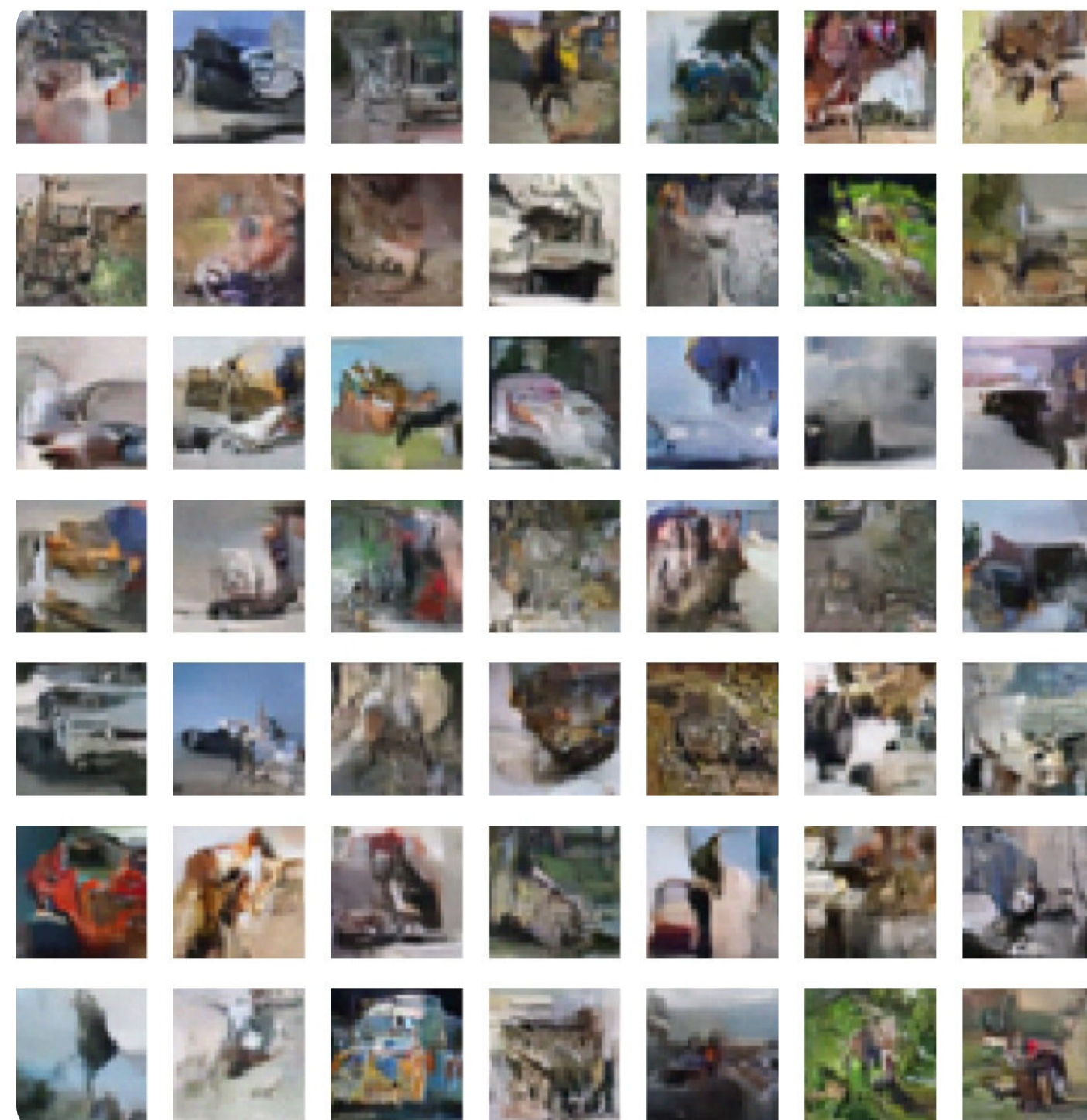
$$q(x_t | x_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} x_{t-1}, \beta_t I)$$

shrink a little, add a little noise, repeat

Tiny steps $\Rightarrow$ the **reverse** step is $\approx$ Gaussian too
(Feller, 1949) $\Rightarrow$ fit it with an **ELBO**

$\rightarrow$ Destroy data slowly with a fixed process, then learn to undo each tiny step: the whole idea, in 2015.

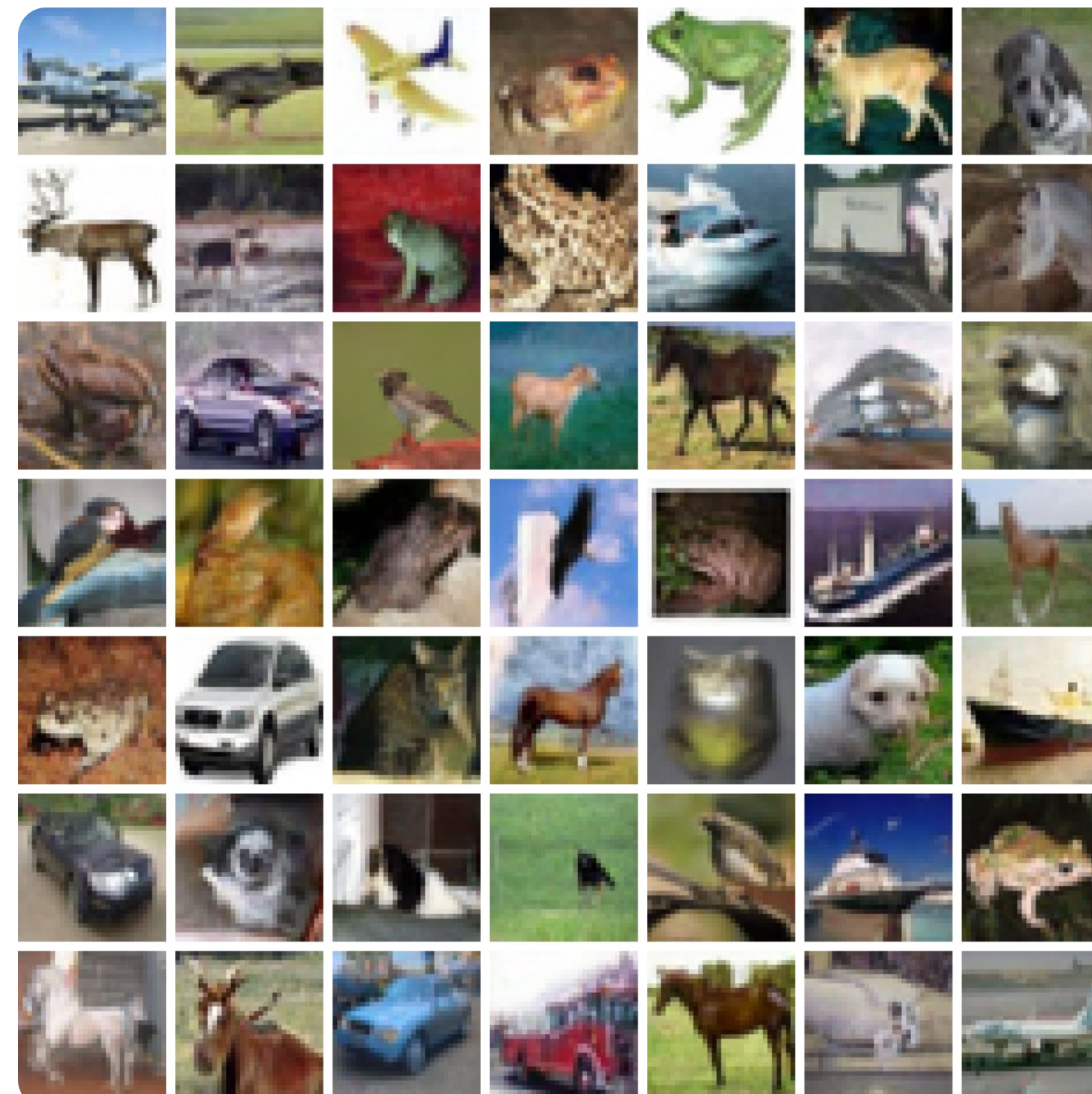
A 2015 idea, then **five quiet years**. Why?



Sohl-Dickstein et al., 2015 · Fig. 3(d)

2015 · CIFAR-10 samples

vs



Ho et al., 2020 · Fig. 13 (7×7 crop)

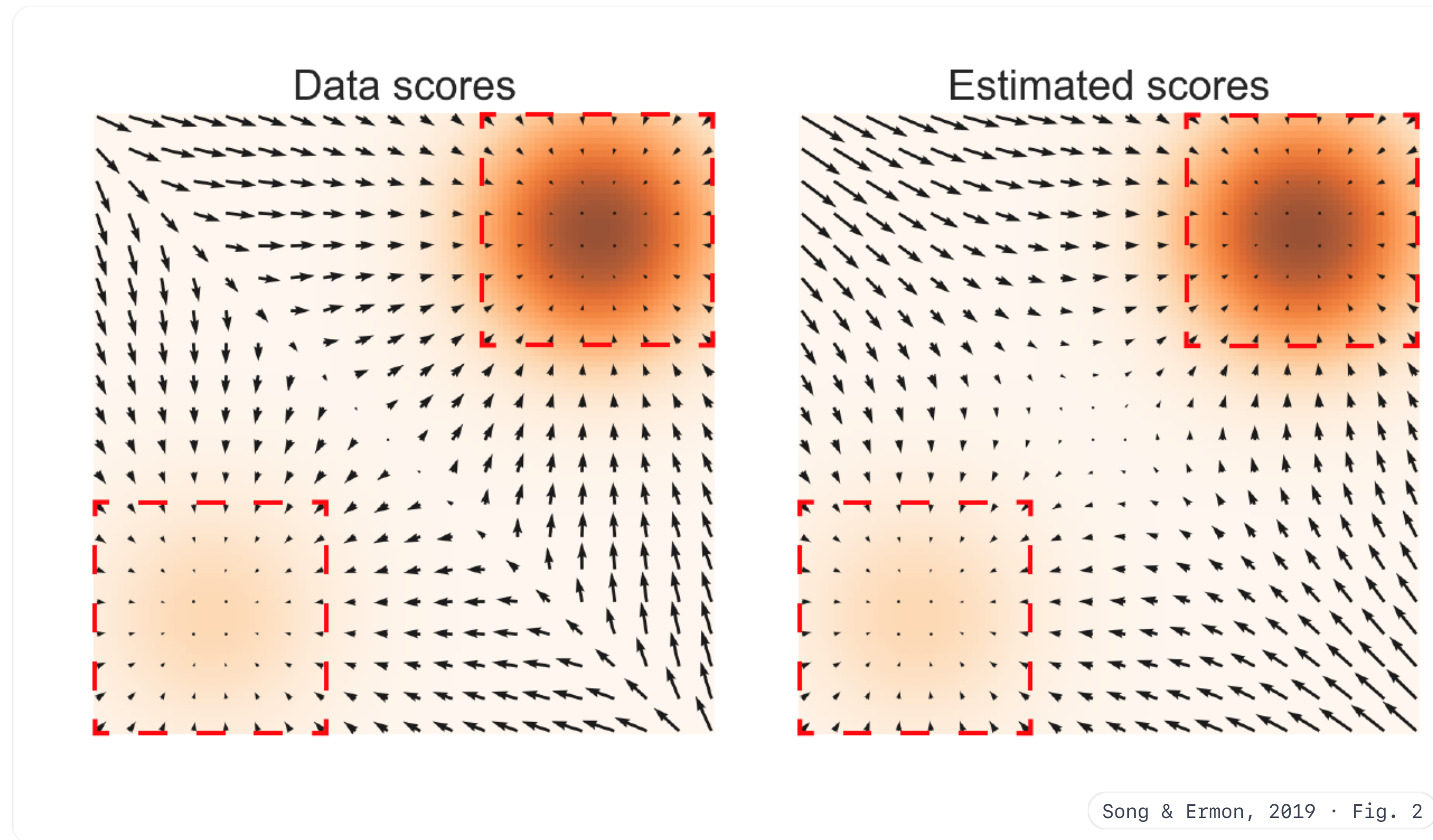
2020 · DDPM, CIFAR-10

speaker interpretation

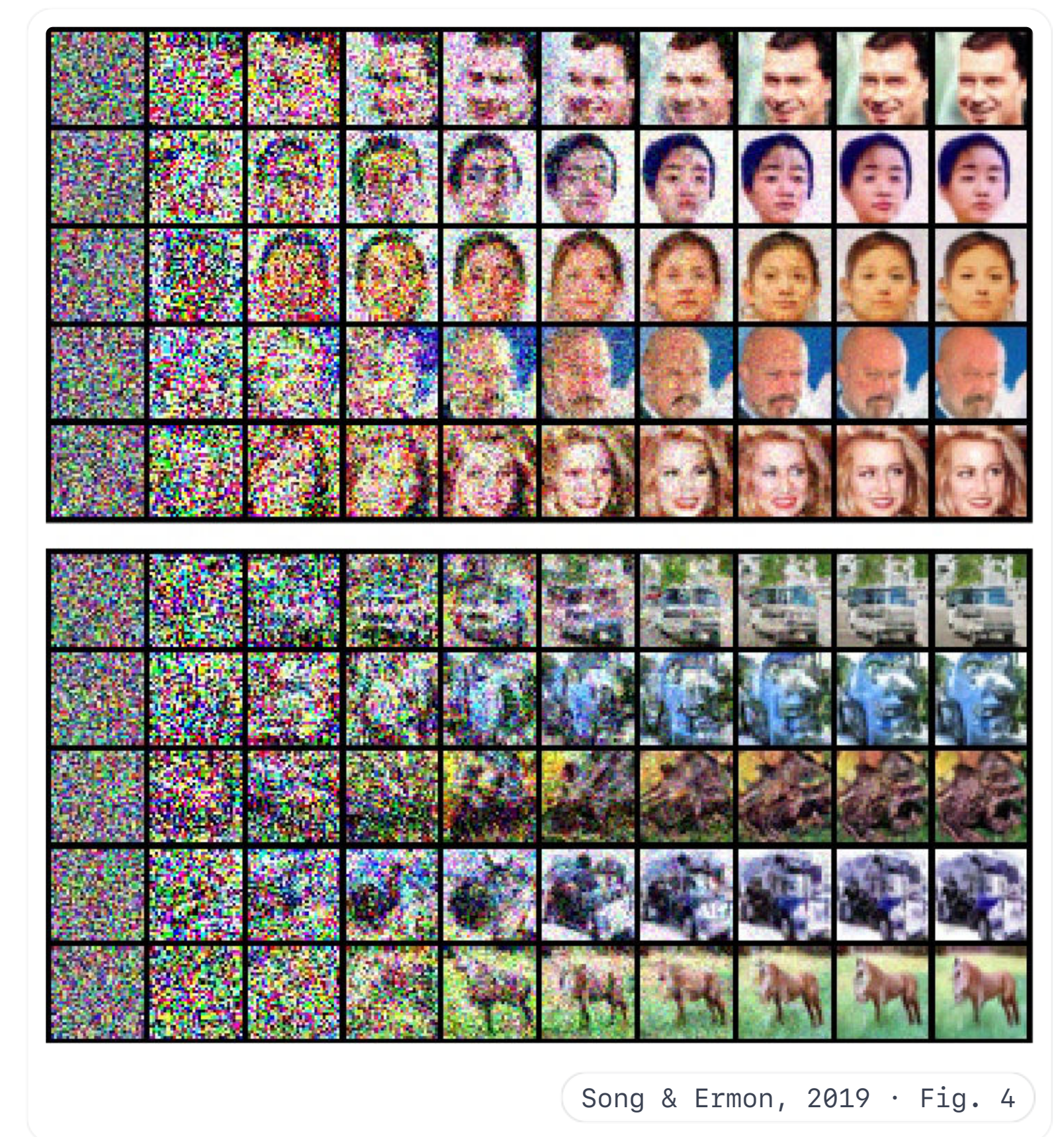
- **Quality:** far behind GANs
- **No recipe yet:** ϵ -prediction, simple loss, attention U-Net
- **Speed:** hundreds to thousands of steps vs one GAN pass

→ One reading: the 2015 idea was right but lacked a training recipe; DDPM supplied one. Speed came later.

NCSN: add noise **on purpose**, at many scales



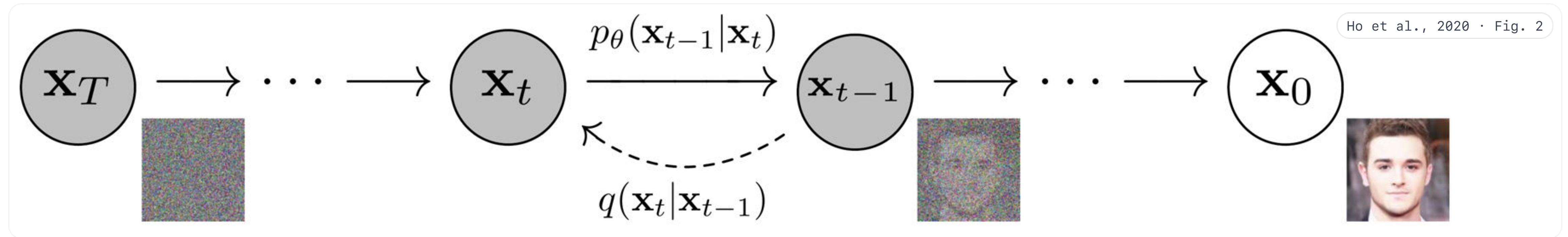
Problem: learned scores are only right near the data (boxes)



Fix: anneal σ from 1 $\rightarrow$ 0.01

→ Big noise fills empty space so the score is learnable everywhere; then shrink the noise, coarse to fine.

DDPM: the 2015 chain + one regression loss



THE KEY SIMPLIFICATION

$$-\text{ELBO} \simeq \sum_t w_t \mathbb{E} \|\epsilon - \epsilon_\theta(x_t, t)\|^2 \xrightarrow{w_t \rightarrow 1} L_{\text{simple}}$$

each term = denoising at one noise level; set all weights to 1

RECIPE REUSED FOR YEARS

$T = 1000$ linear β : $10^{-4} \rightarrow 0.02$

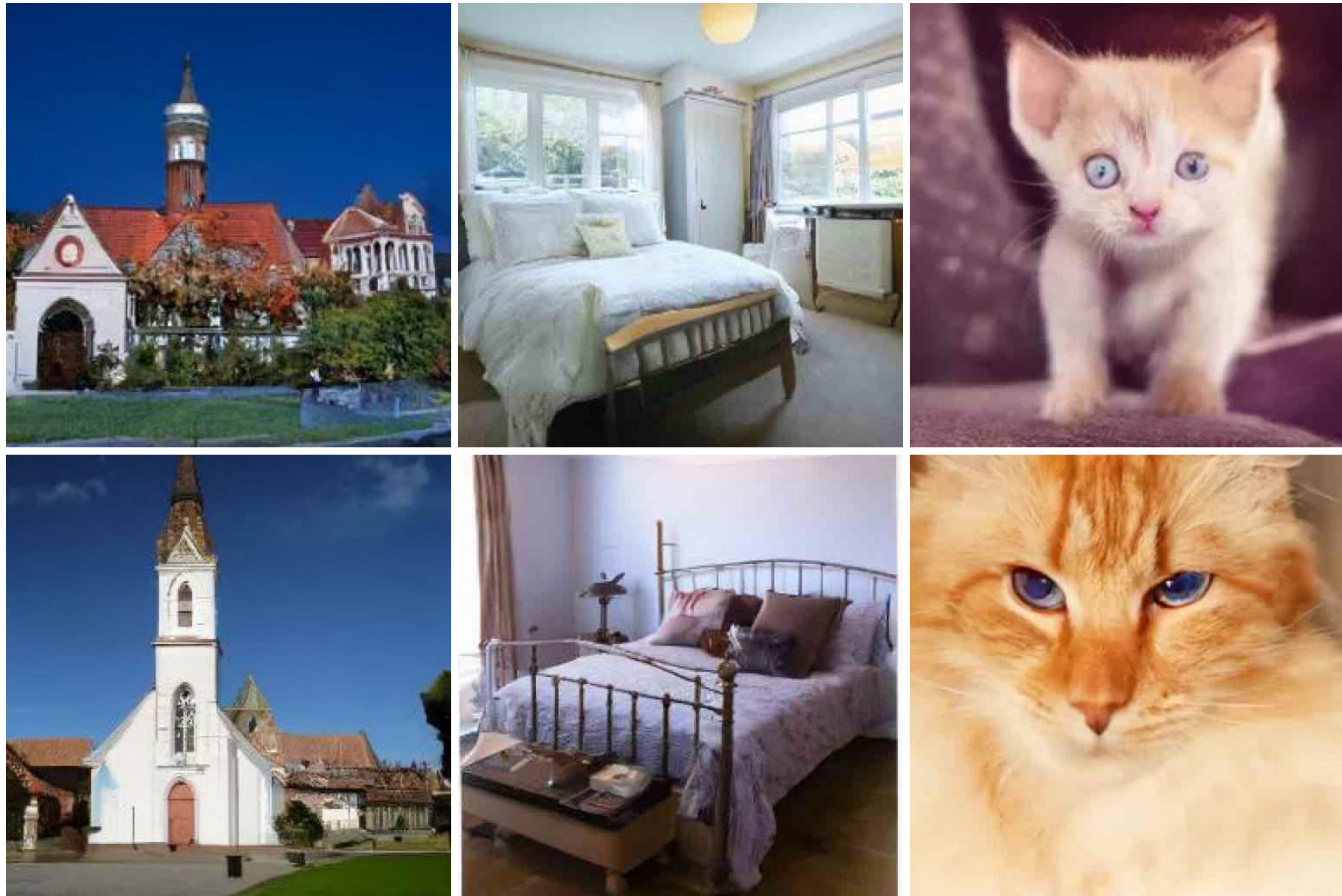
U-Net + attention at 16^2

fixed variance

sinusoidal t

→ DDPM showed the 2015 objective is weighted noise regression, dropped the weights, and added a modern U-Net.

It finally works: GAN-level images, no GAN



3.17

FID · CIFAR-10 (LOWER IS BETTER)

9.46

INCEPTION SCORE · CIFAR-10

LSUN 256²: comparable to **ProgressiveGAN**, with no discriminator

LSUN 256² · church · bedroom · cat

Samples: Ho et al., 2020 · Figs. 16, 17, 19 (tiles)

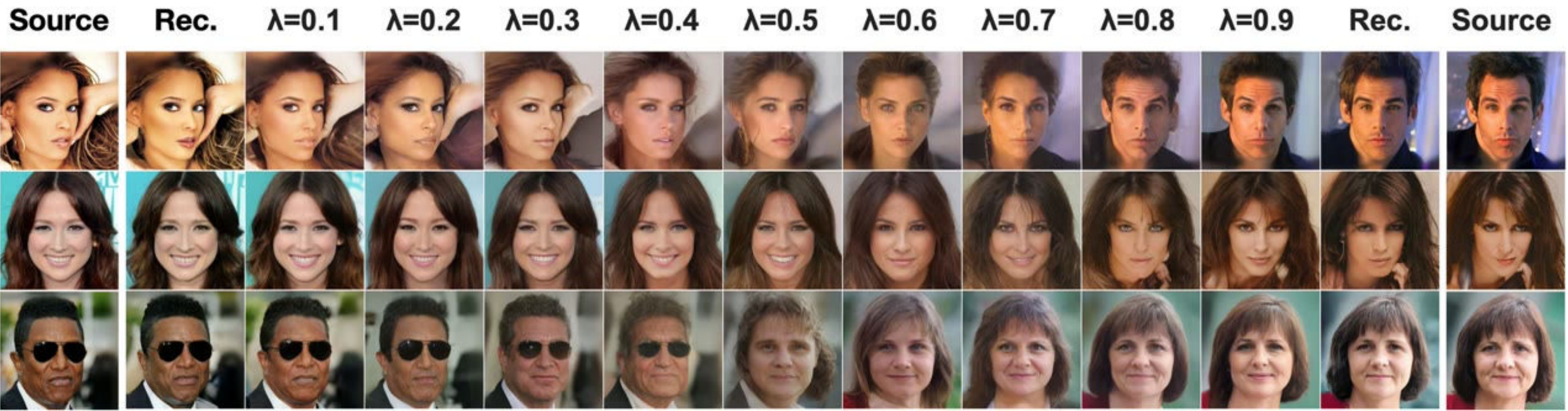
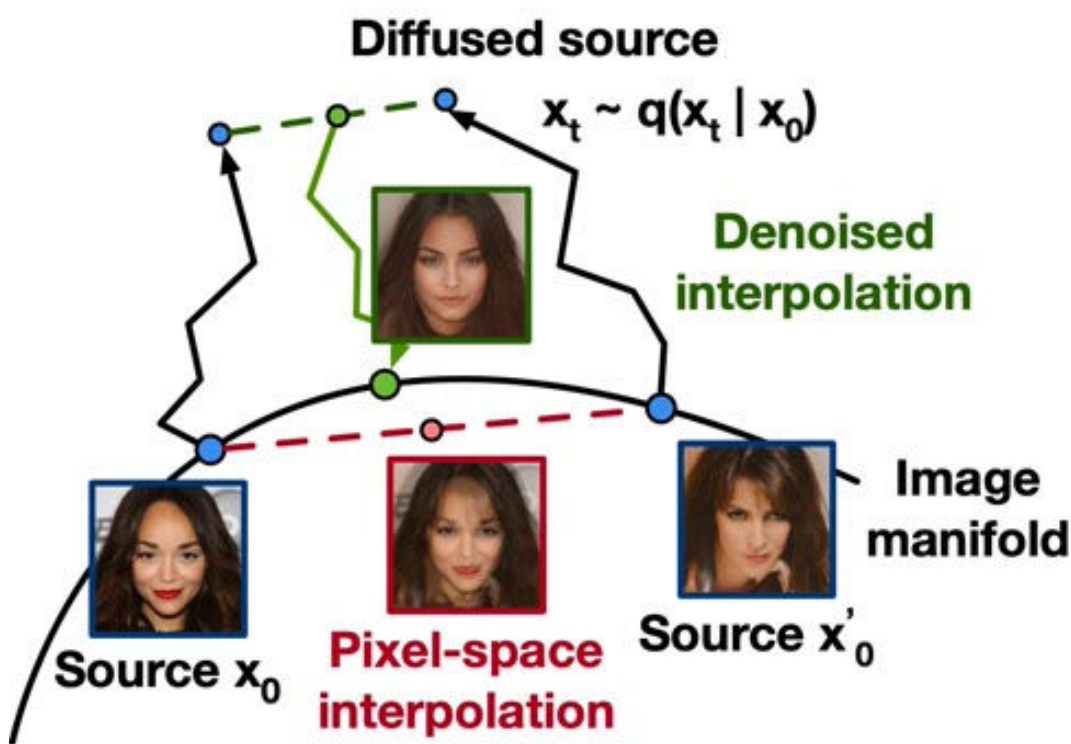
→ First diffusion samples that could stand next to GANs, from a network trained with one MSE loss.

Noise level = level of abstraction



Share x_t , resample the rest: high t keeps only coarse attributes

Ho et al., 2020 · Fig. 7

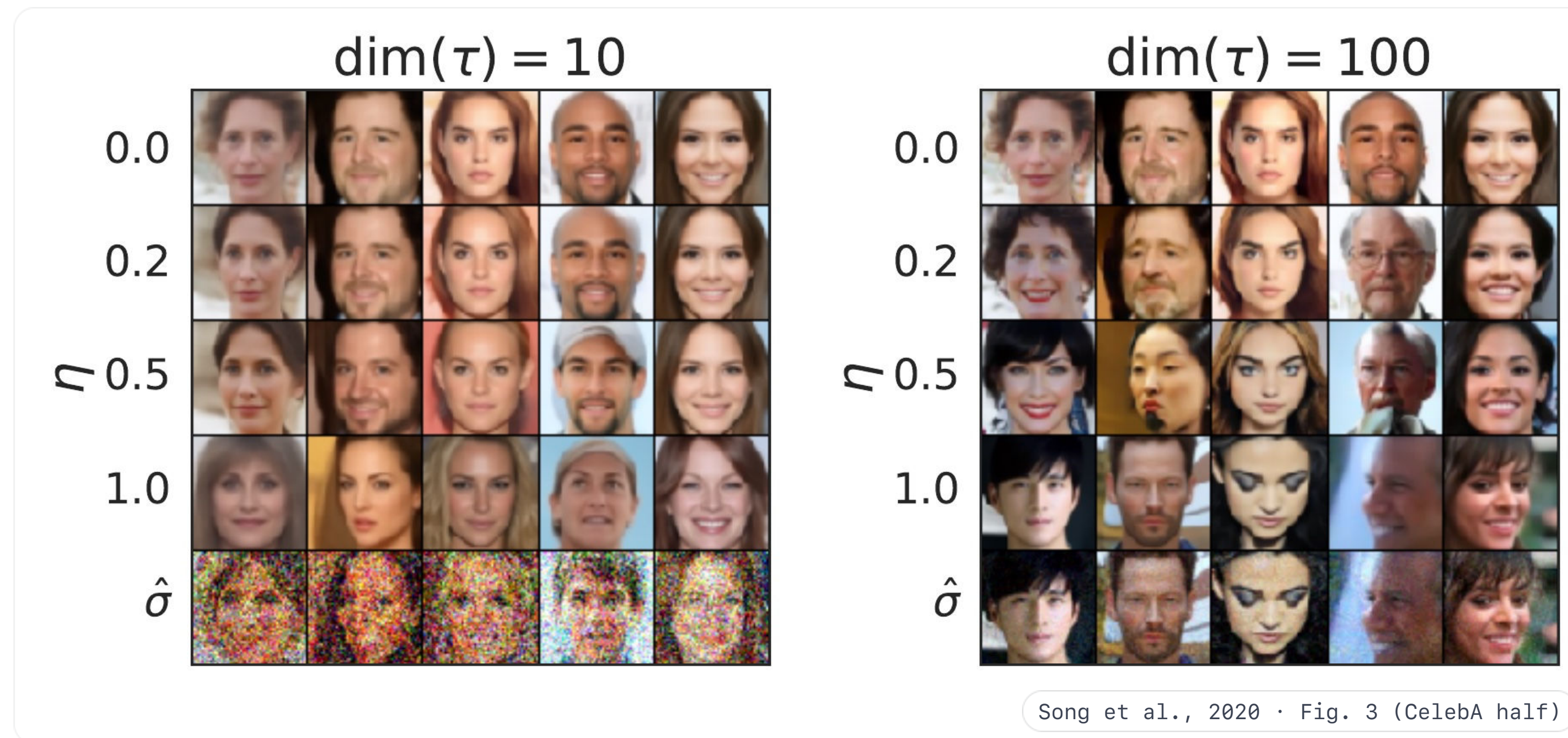


Noise both, mix, denoise: smooth faces, unlike pixel blending

Ho et al., 2020 · Fig. 8

→ Noisy early steps decide coarse attributes, late steps decide fine detail: diffusion generates coarse to fine.

DDIM: **same network**, no randomness, far fewer steps



10 vs 100 steps · rows $\eta = 0 \rightarrow 1$ · bottom $\hat{\sigma}$: DDPM, larger variance

ONE DDIM STEP ($\eta = 0$)

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \hat{x}_0 + \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_{\theta}(x_t, t)$$

① predict clean $\hat{x}_0$ ② re-noise it with the same ϵ

10–50×

FASTER, NO RETRAINING

50

DDIM STEPS: ORIGINAL SD SCRIPT

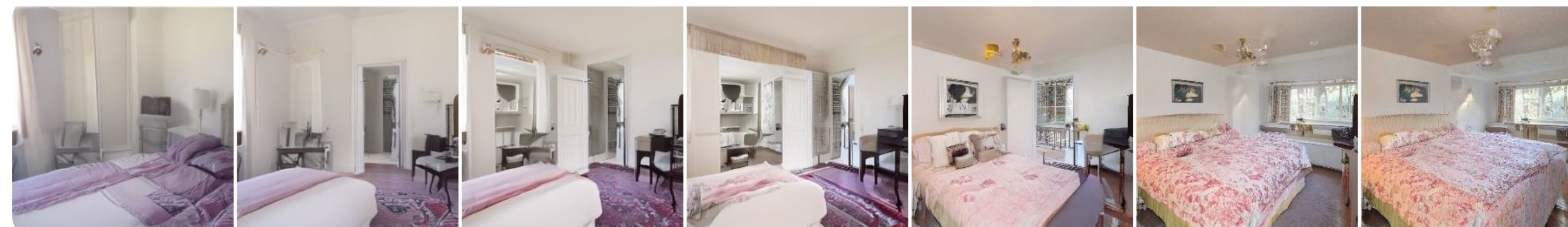
→ Keep the trained DDPM, change only the sampler: deterministic steps let you skip most of the chain.

Fixed noise → fixed image: **invert and edit**



Same x_T , 20 → 1000 steps ≈ same face

Song et al., 2020 · Fig. 9 (columns 9–16, adapted)

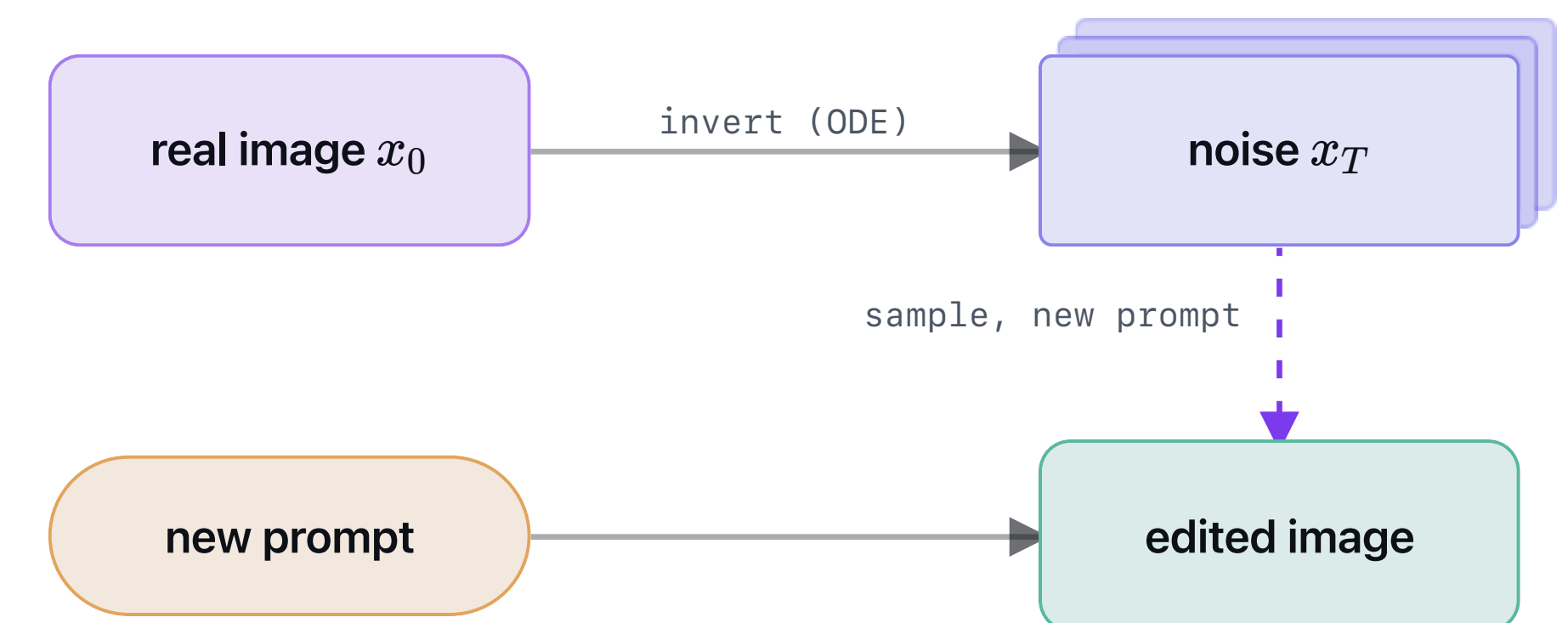


Interpolate x_T : smooth semantic morph

Song et al., 2020 · Fig. 6 (bedroom row)

→ With no randomness, noise and image map one-to-one, so a real photo can be encoded, then re-generated.

DDIM INVERSION

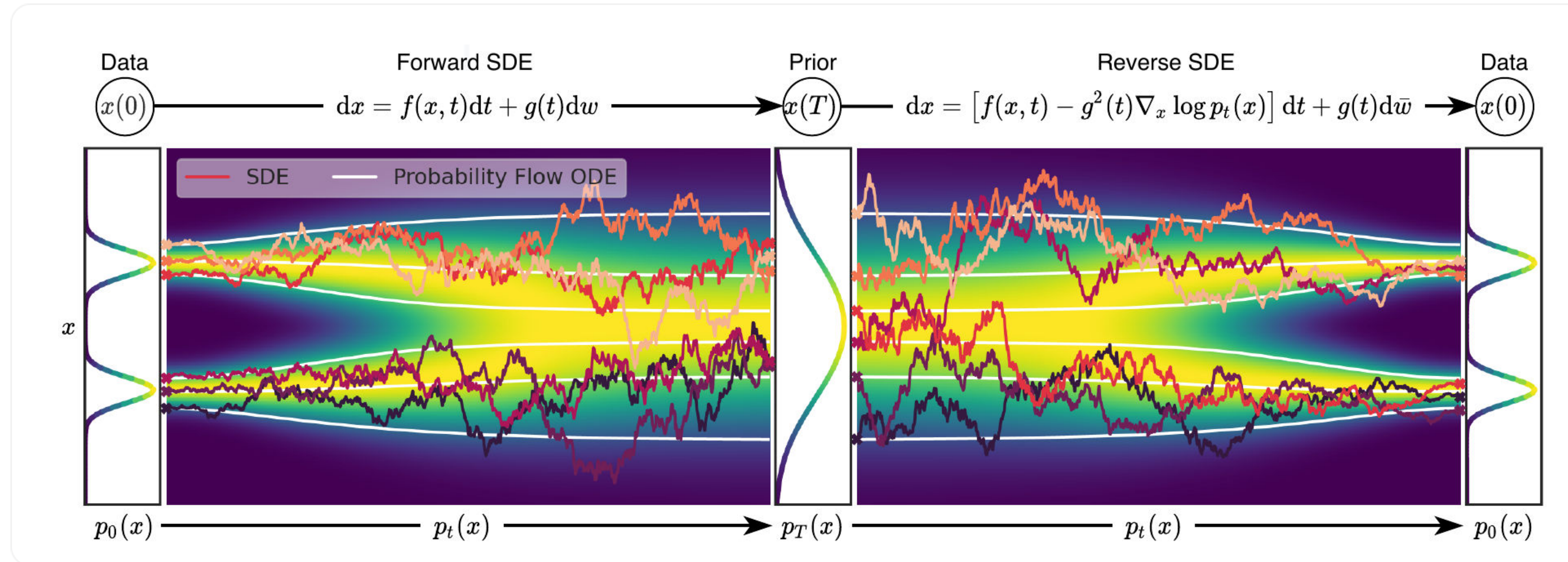


Prompt-to-Prompt 2022.08

Null-text Inversion 2022.11

approximate: error grows with large CFG

Score SDE: **one equation** for NCSN and DDPM



Song et al., 2020 · Fig. 2

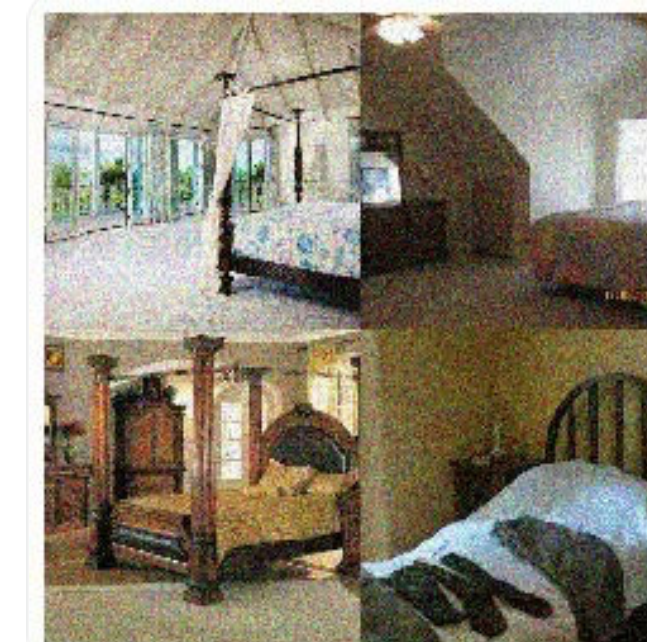
PROBABILITY-FLOW ODE · WHITE LINES

$$\frac{dx}{dt} = f(x, t) - \frac{1}{2} g(t)^2 \nabla_x \log p_t(x)$$

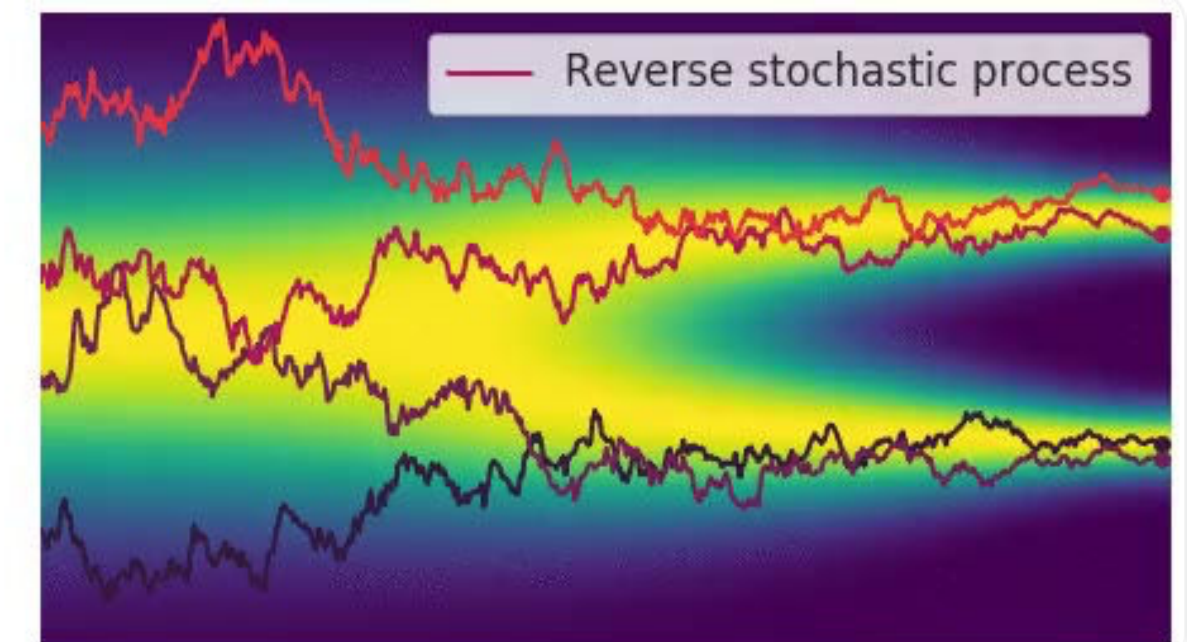
same marginals as the SDE, no noise: exact likelihood, invertible

NCSN = variance-exploding (VE)

DDPM = variance-preserving (VP)



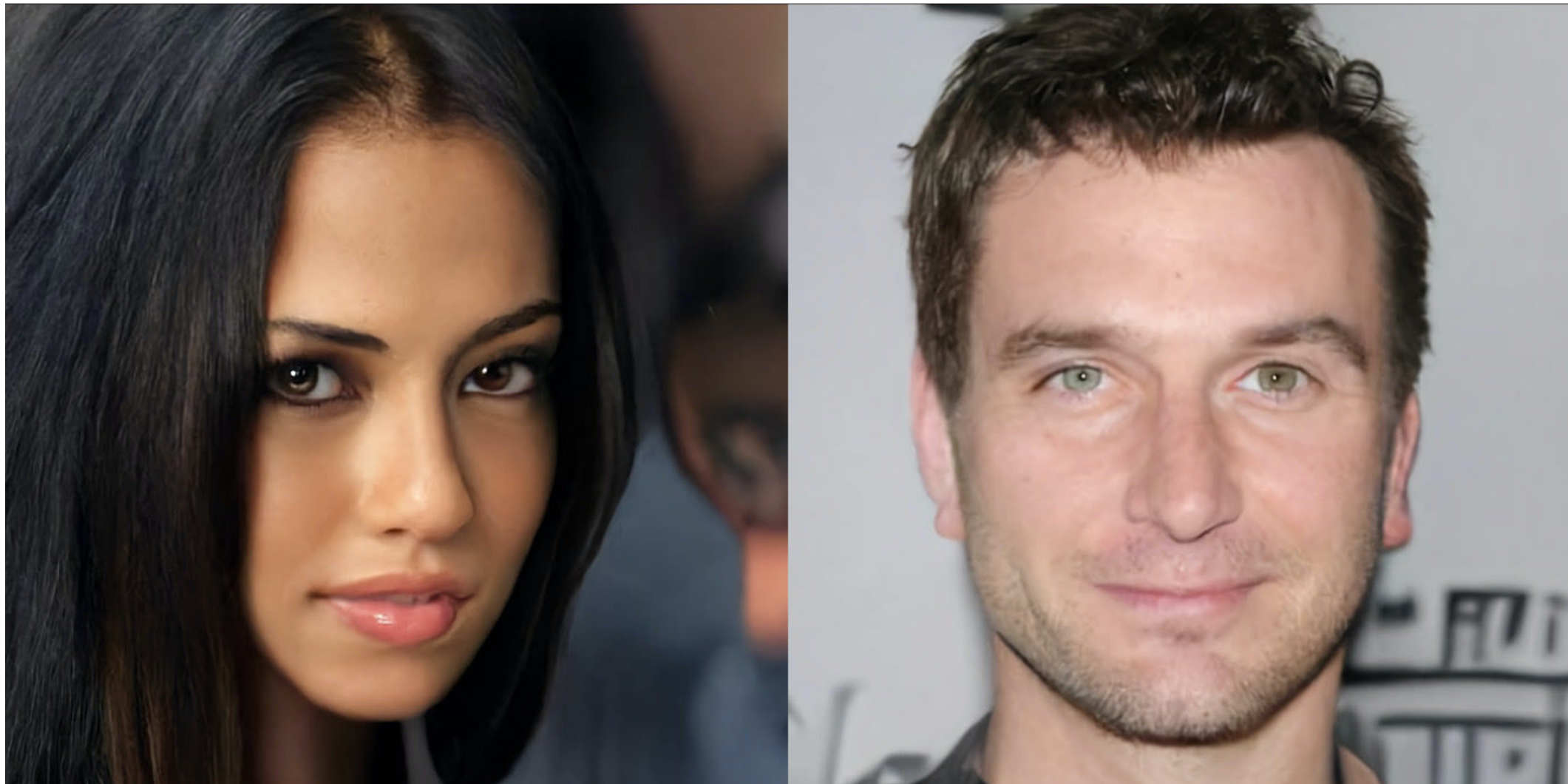
Reverse SDE



Video: Yang Song, score-based models blog (2021)

→ Pick a noising SDE; running it backwards needs only the score, which a denoiser already learns.

CIFAR records, 1024² faces, no-retrain editing



1024² CelebA-HQ: a first for score-based models

Samples: Song et al., 2020 · Fig. 12 (2 samples, adapted)



Inpaint (top), colorize (bottom) · col 1: original, col 2: input

Song et al., 2020 · Fig. 4 (right)

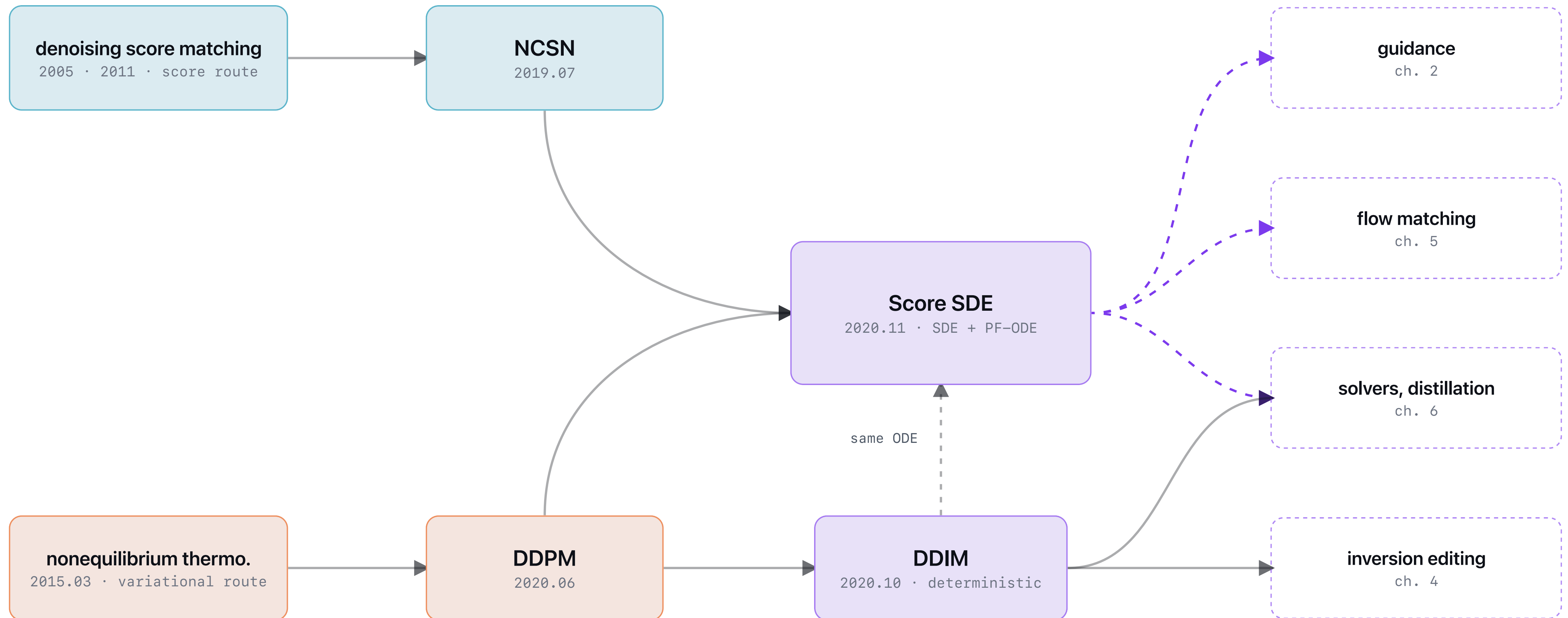
CIFAR-10 FID ↓ · UNCONDITIONAL · LOWER IS BETTER



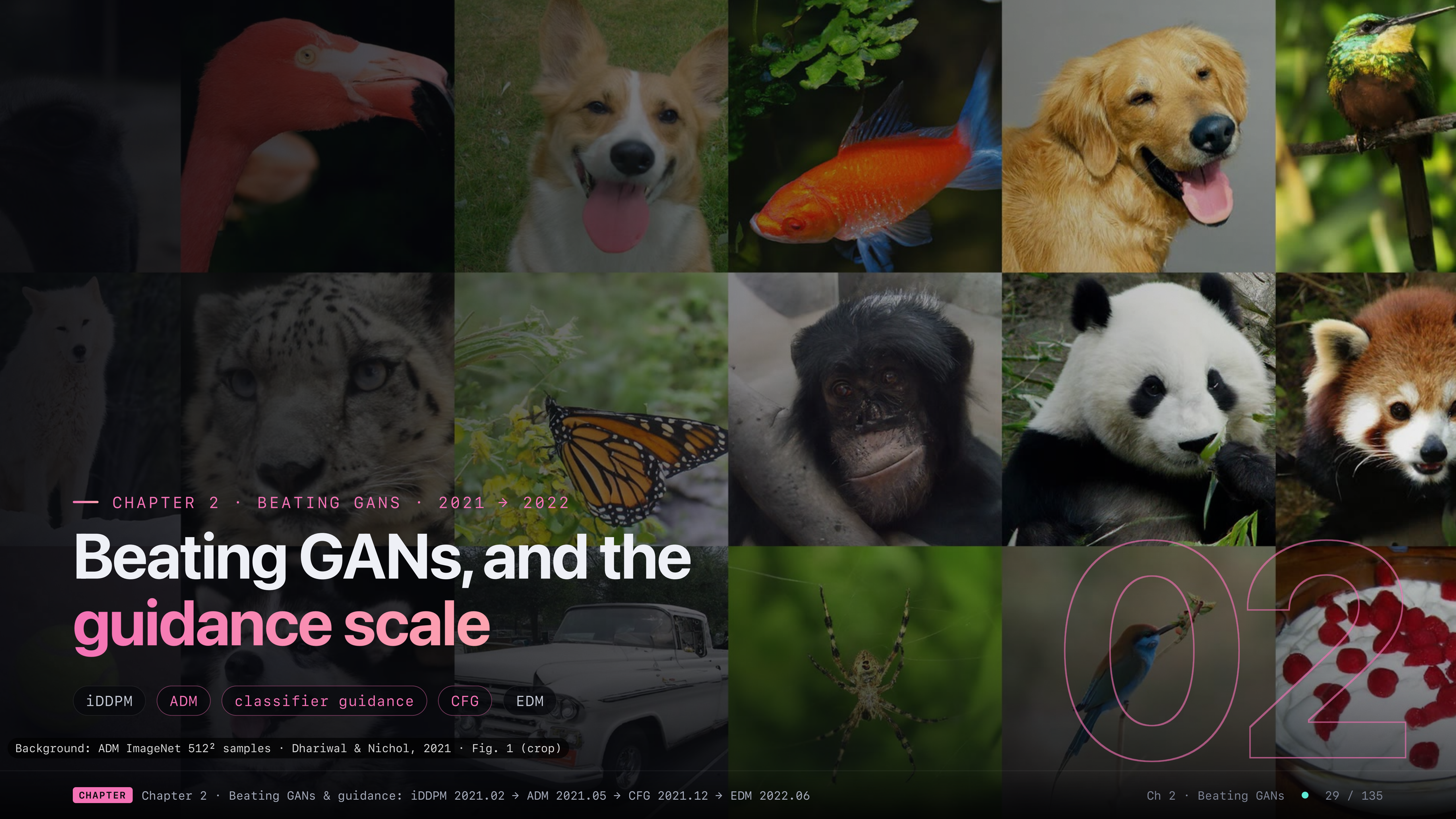
numbers as reported in each paper

→ One score model, many tasks: conditions are added at sampling time, so inpainting needs no retraining.

Two routes, one framework



→ By the end of 2020: one framework, one recipe, CIFAR records. Not yet: beating GANs on ImageNet, or speed.



— CHAPTER 2 · BEATING GANS · 2021 → 2022

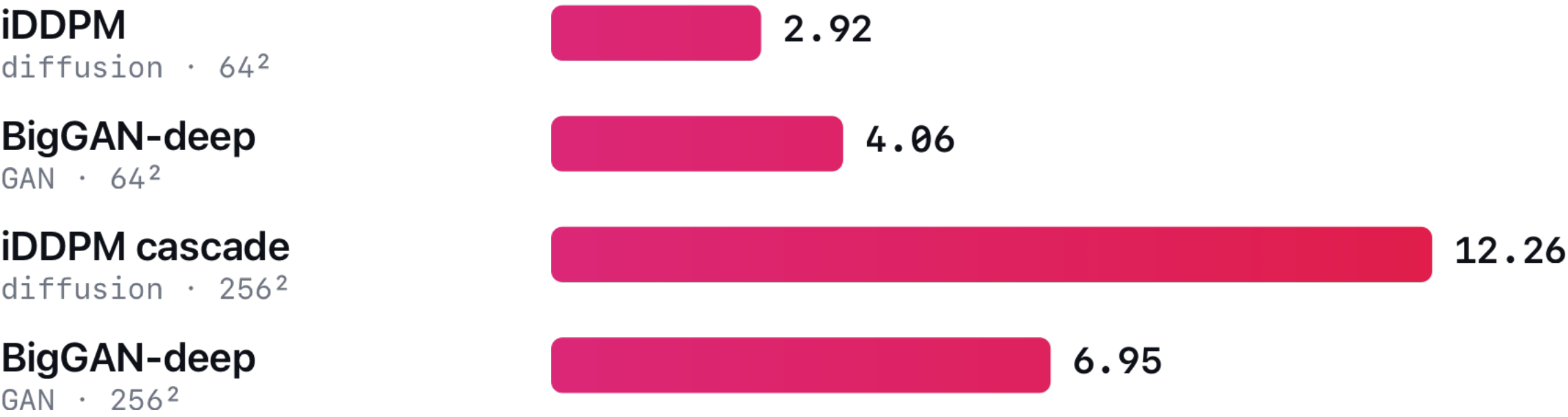
Beating GANs, and the guidance scale

iDDPM ADM classifier guidance CFG EDM

Background: ADM ImageNet 512² samples · Dhariwal & Nichol, 2021 · Fig. 1 (crop)

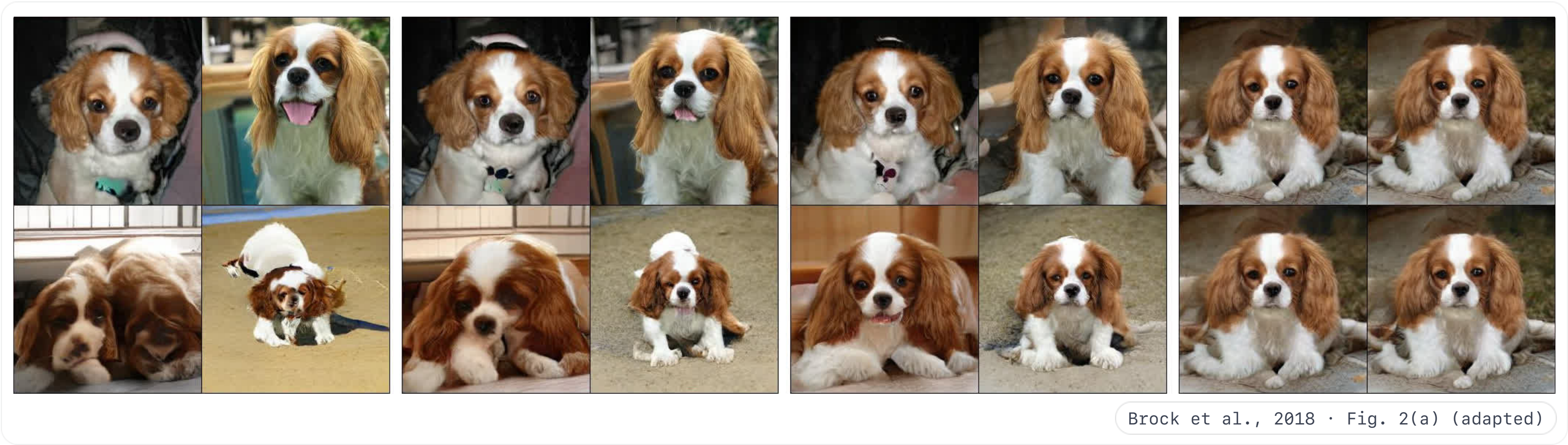
Diffusion wins at 64², but GANs still own 256²

IMAGENET FID ↓



DIFFUSION LACKED

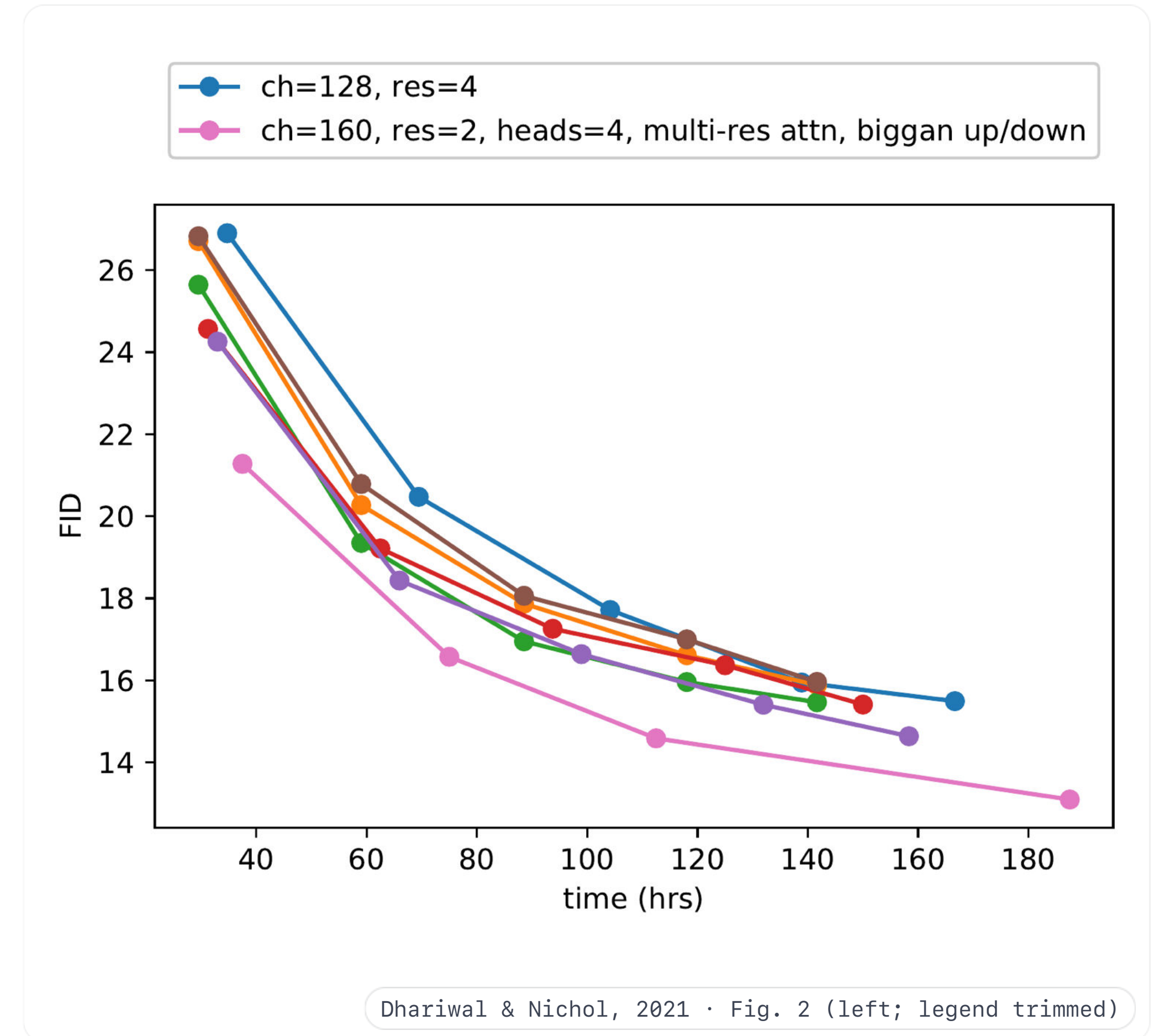
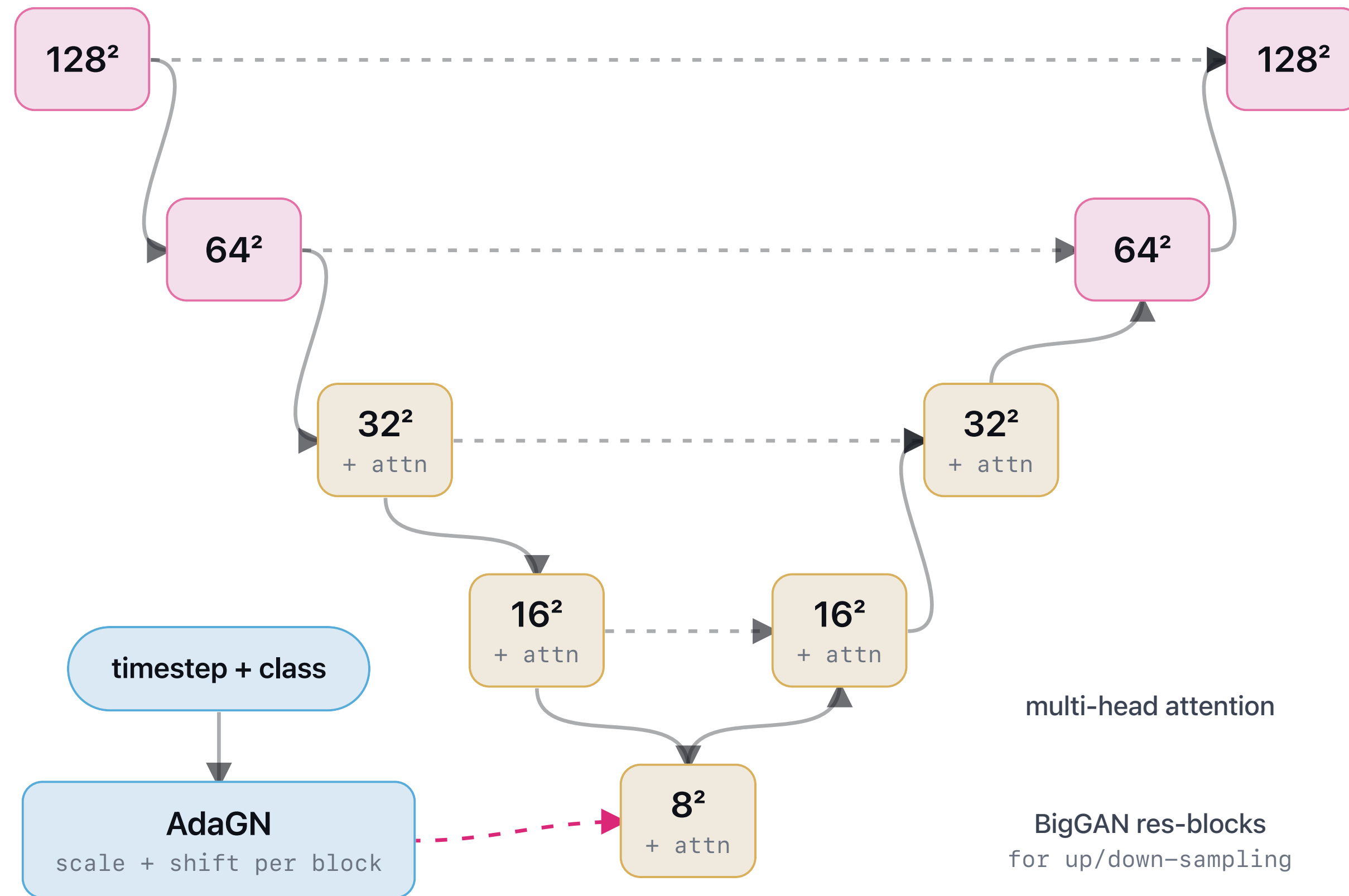
- ① a systematically ablated U-Net
- ② a diversity ↔ fidelity control



GAN truncation: noise nearer the centre → typical, less varied

➔ Diffusion led on small images; GANs led at 256² and had a fidelity control.

Fix 1: a U-Net tuned by systematic ablation



pink = helpful changes combined: lowest FID throughout

→ No single trick: more attention, BigGAN-style blocks, and timestep + class steering every block.

Fix 2: a classifier **steers** every denoising step



$s = 1$ · unconditional model, guided to "corgi"

Dhariwal & Nichol, 2021 · Fig. 3 (left)



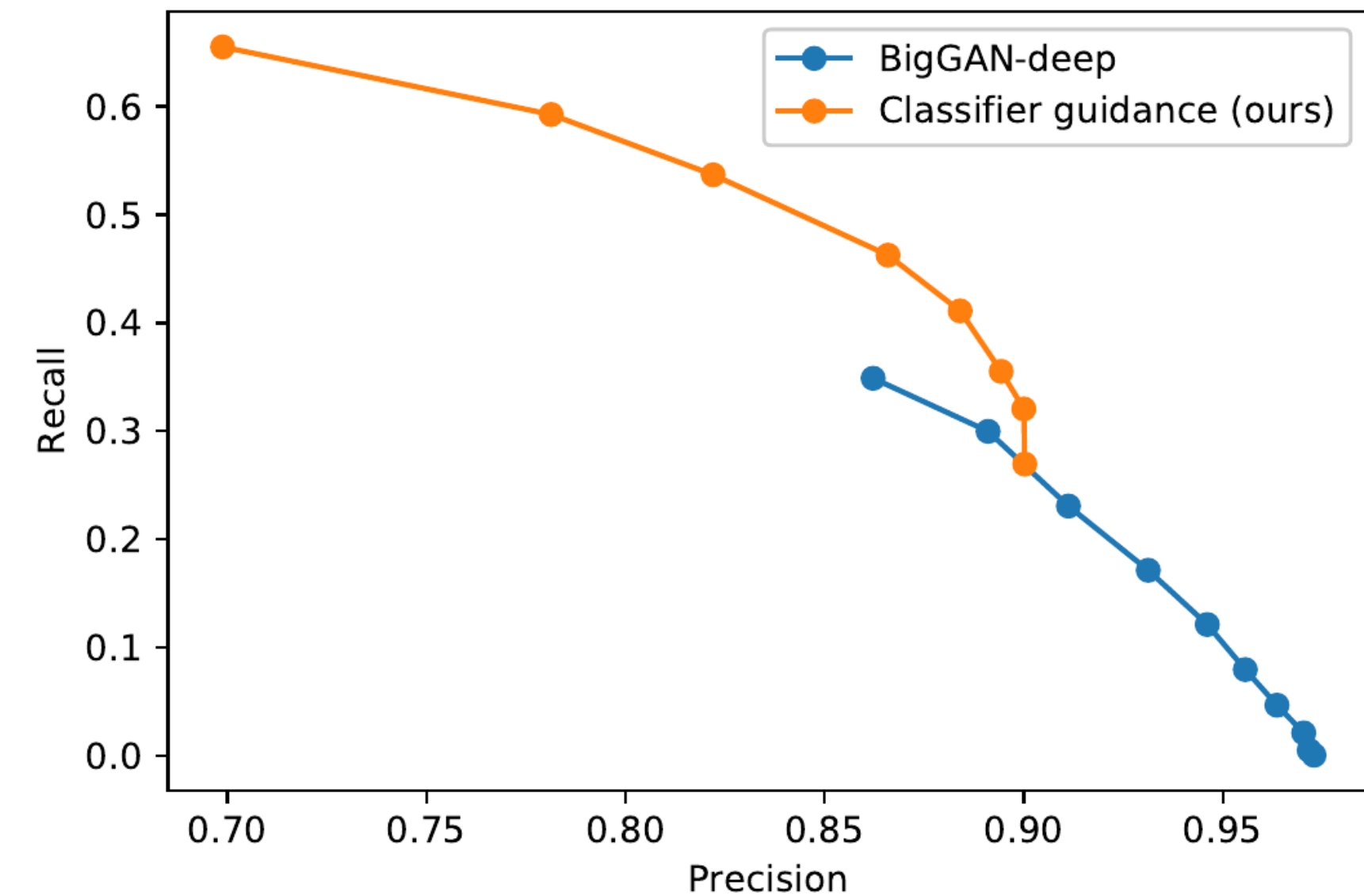
$s = 10$ · clearly corgis, and more alike

Dhariwal & Nichol, 2021 · Fig. 3 (right)

GUIDED SCORE

$$\nabla \log \tilde{p}(x_t|y) = \nabla \log p_\theta(x_t) + s \nabla \log p_\phi(y|x_t)$$

model's direction + $s \times$ "look more like class y "



Dhariwal & Nichol, 2021 · Fig. 5 (left)

orange guidance keeps more recall than **blue** GAN truncation (noise near centre)

→ A classifier nudges every step toward the class; its volume s trades variety (recall) for fidelity (precision).

Diffusion **beats BigGAN**, and covers more modes



BigGAN-deep · FID 6.95

Dhariwal & Nichol, 2021 · Fig. 6 (left, flamingo rows, adapted)



ADM-G, s = 1 · FID 4.59

Dhariwal & Nichol, 2021 · Fig. 6 (middle, flamingo rows, adapted)

4.59

FID ↓ · BIGGAN-DEEP 6.95

0.52

RECALL ↑ · BIGGAN-DEEP 0.28

precision: BigGAN-deep still higher, 0.87 vs 0.82

no guidance 10.94 · + upsampler 3.94 · 25 steps 5.44

→ First diffusion model to beat the best GAN on ImageNet 256², with far more varied samples.

Drop the classifier: **one network** guides itself

Classifier guidance: ① extra noise-aware classifier
② no open-text labels · ③ adversarial-like gradients

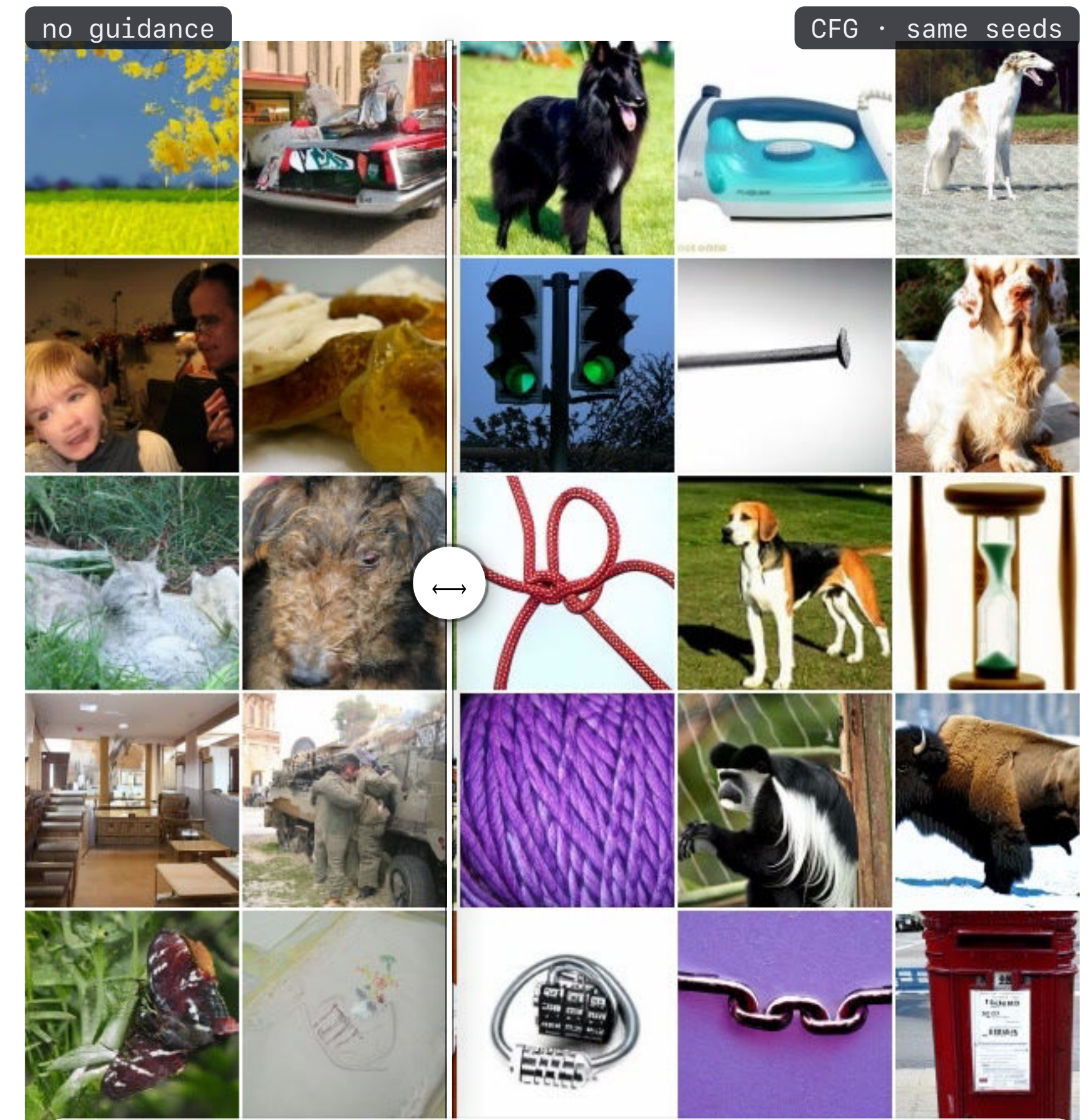
BAYES · IMPLICIT CLASSIFIER

$$\nabla \log p(c|x) = \nabla \log p(x|c) - \nabla \log p(x)$$

conditional – unconditional pass = its gradient

Recap: $\varepsilon(x, \emptyset) + w \cdot (\varepsilon(x, c) - \varepsilon(x, \emptyset))$, c dropped 10–20%

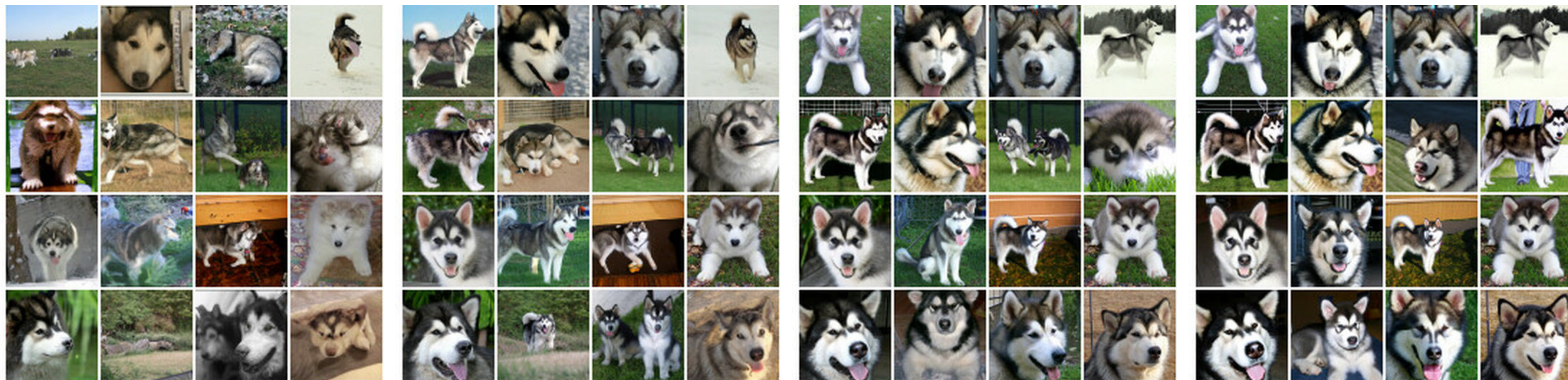
$\emptyset \rightarrow$ negative prompt



Ho & Salimans, 2021 · arXiv Fig. 7(a) vs 7(c), left halves (adapted)

→ One network, two passes: their gap is the guidance, no classifier needed.

Turn up w: **sharper classes**, less variety



Ho & Salimans, 2021 · arXiv Fig. 1 (upscaled)

Malamute, ImageNet 64²: no guidance → stronger guidance

1.55

BEST FID · IMAGENET 64² · AT w = 1.1

Inception Score keeps rising · FID falls, then rises

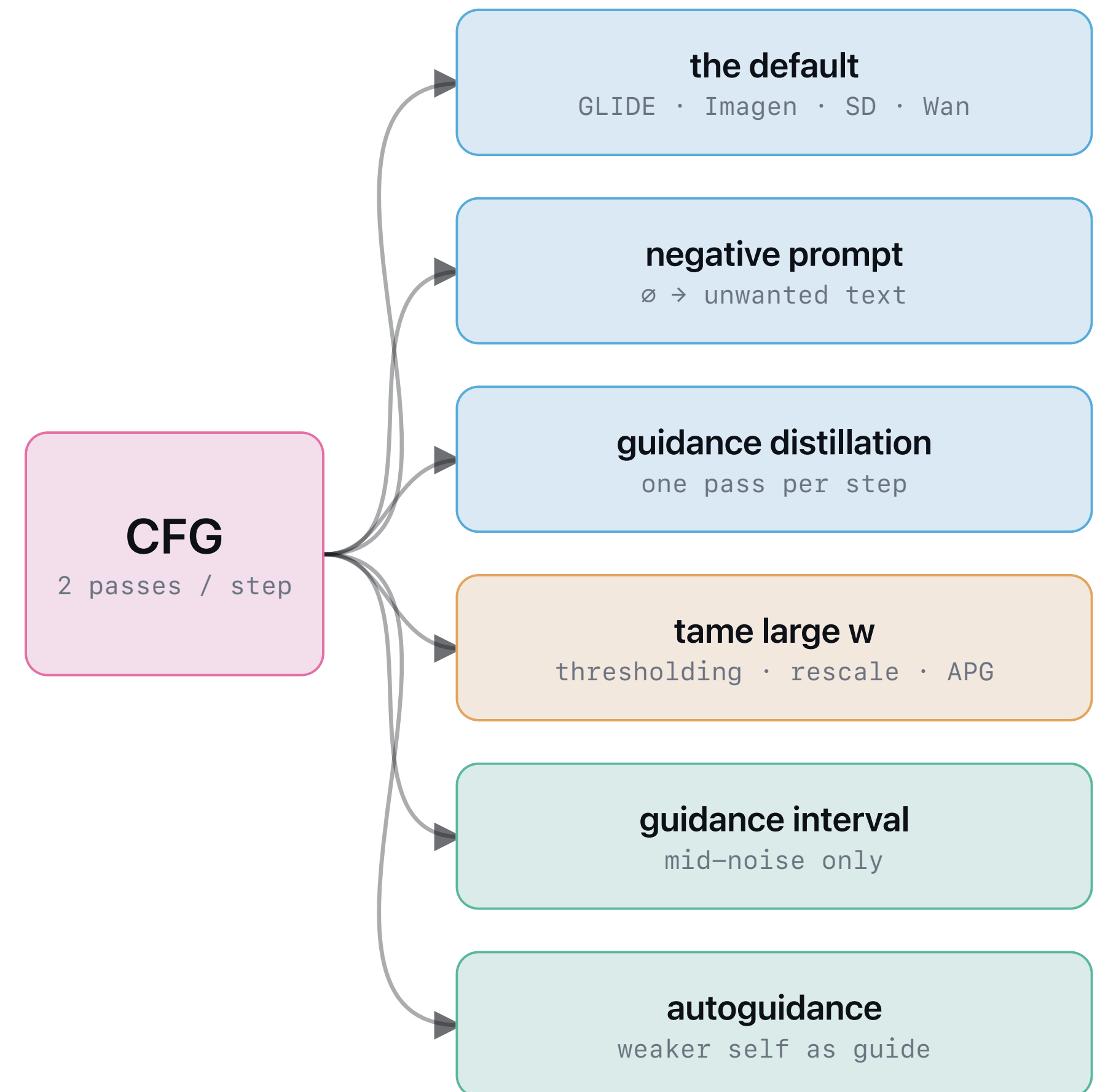
large w: saturated colours, lost diversity

→ More guidance makes samples more typical and polished, and also more alike.

CFG became the default, and a **research field**



left, CFG: whole branches emptied · **right, autoguidance:** variety kept



→ Two passes per step became the norm; follow-ups make it cheaper, or keep the variety.

EDM: one noise level σ , independent choices

	VP [49]	VE [49]	iDDPM [37] + DDIM [47]	Ours (“EDM”)
Sampling (Section 3)				
ODE solver	Euler	Euler	Euler	2 nd order Heun
Time steps	$t_{i < N}$	$1 + \frac{i}{N-1}(\epsilon_s - 1)$	$\sigma_{\max}^2 (\sigma_{\min}^2 / \sigma_{\max}^2)^{\frac{i}{N-1}}$	$(\sigma_{\max}^{\frac{1}{\rho}} + \frac{i}{N-1}(\sigma_{\min}^{\frac{1}{\rho}} - \sigma_{\max}^{\frac{1}{\rho}}))^\rho$
Schedule	$\sigma(t)$	$\sqrt{e^{\frac{1}{2}\beta_d t^2 + \beta_{\min} t} - 1}$	t	t
Scaling	$s(t)$	$1/\sqrt{e^{\frac{1}{2}\beta_d t^2 + \beta_{\min} t}}$	1	1
Network and preconditioning (Section 5)				
Architecture of F_θ	DDPM++	NCSN++	DDPM	(any)
Skip scaling $c_{\text{skip}}(\sigma)$	1	1	1	$\sigma_{\text{data}}^2 / (\sigma^2 + \sigma_{\text{data}}^2)$
Output scaling $c_{\text{out}}(\sigma)$	$-\sigma$	σ	$-\sigma$	$\sigma \cdot \sigma_{\text{data}} / \sqrt{\sigma_{\text{data}}^2 + \sigma^2}$
Input scaling $c_{\text{in}}(\sigma)$	$1/\sqrt{\sigma^2 + 1}$	1	$1/\sqrt{\sigma^2 + 1}$	$1/\sqrt{\sigma^2 + \sigma_{\text{data}}^2}$
Noise cond. $c_{\text{noise}}(\sigma)$	$(M - 1) \sigma^{-1}(\sigma)$	$\ln(\frac{1}{2}\sigma)$	$M - 1 - \arg \min_j u_j - \sigma $	$\frac{1}{4} \ln(\sigma)$
Training (Section 5)				
Noise distribution	$\sigma^{-1}(\sigma) \sim \mathcal{U}(\epsilon_t, 1)$	$\ln(\sigma) \sim \mathcal{U}(\ln(\sigma_{\min}), \ln(\sigma_{\max}))$	$\sigma = u_j, j \sim \mathcal{U}\{0, M - 1\}$	$\ln(\sigma) \sim \mathcal{N}(P_{\text{mean}}, P_{\text{std}}^2)$
Loss weighting $\lambda(\sigma)$	$1/\sigma^2$	$1/\sigma^2$	$1/\sigma^2$ (note: *)	$(\sigma^2 + \sigma_{\text{data}}^2) / (\sigma \cdot \sigma_{\text{data}})^2$
Parameters				
	$\beta_d = 19.9, \beta_{\min} = 0.1$	$\sigma_{\min} = 0.02$	$\bar{\alpha}_j = \sin^2(\frac{\pi}{2} \frac{j}{M(C_2+1)})$	$\sigma_{\min} = 0.002, \sigma_{\max} = 80$
	$\epsilon_s = 10^{-3}, \epsilon_t = 10^{-5}$	$\sigma_{\max} = 100$	$C_1 = 0.001, C_2 = 0.008$	$\sigma_{\text{data}} = 0.5, \rho = 7$
	$M = 1000$		$M = 1000, j_0 = 8^\dagger$	$P_{\text{mean}} = -1.2, P_{\text{std}} = 1.2$

Karras et al., 2022 · Table 1 (EDM column highlighted)

every row is a separate choice · pink = EDM

→ EDM writes every diffusion model in one notation, so each design choice can be swapped on its own.

ALL MODELS: DATA + NOISE OF LEVEL σ

$x = y + n, \quad n \sim \mathcal{N}(0, \sigma^2 I)$

$\nabla_x \log p = \frac{D(x; \sigma) - x}{\sigma^2}$

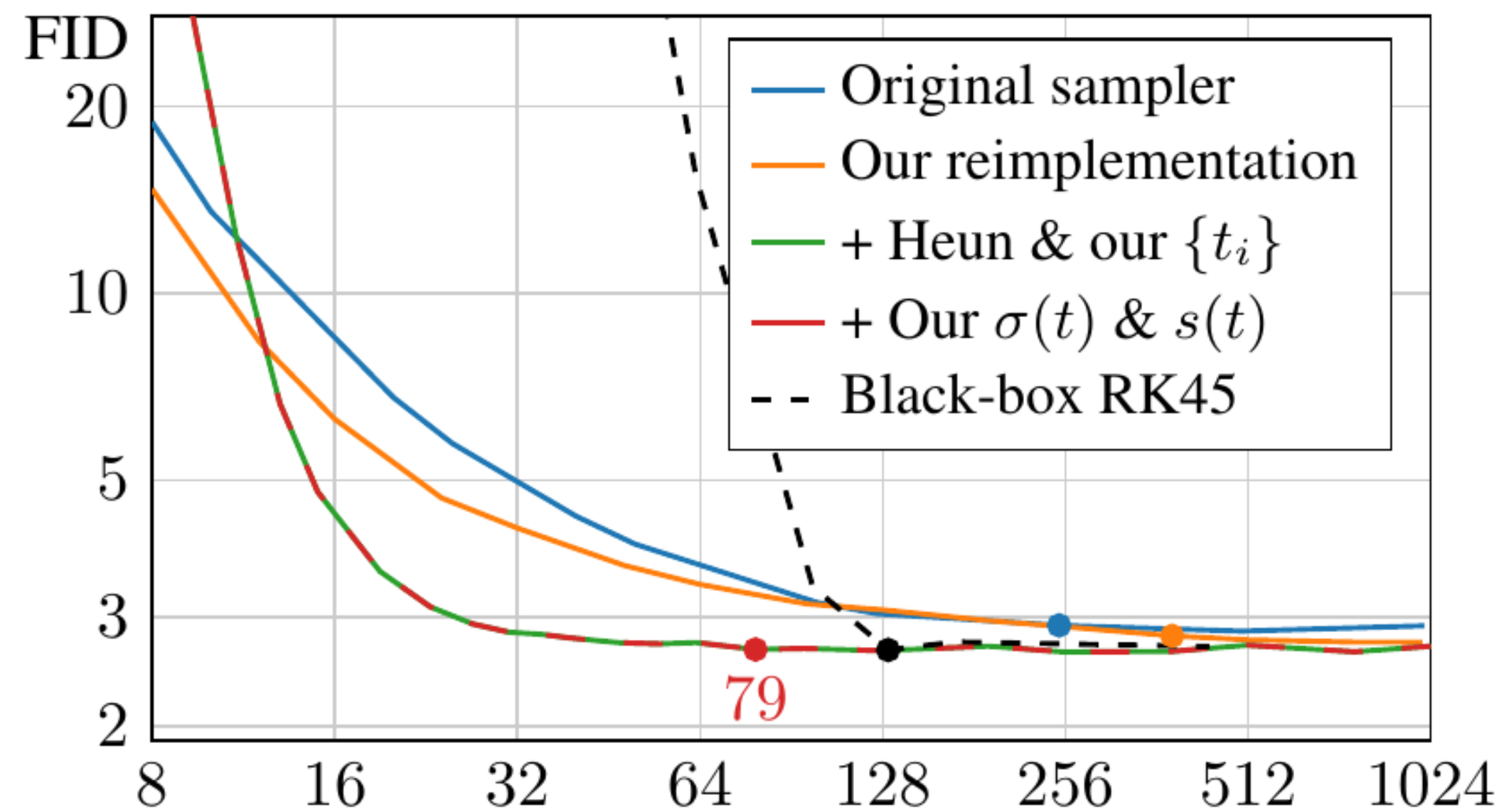
only σ and a denoiser D matter
 σ = primer's σ_t / α_t (noise-to-signal)

(c) DDIM [47] / Our ODE

Karras et al., 2022 · Fig. 3(c)

$\sigma(t) = t$, 1-D toy: mostly straight paths

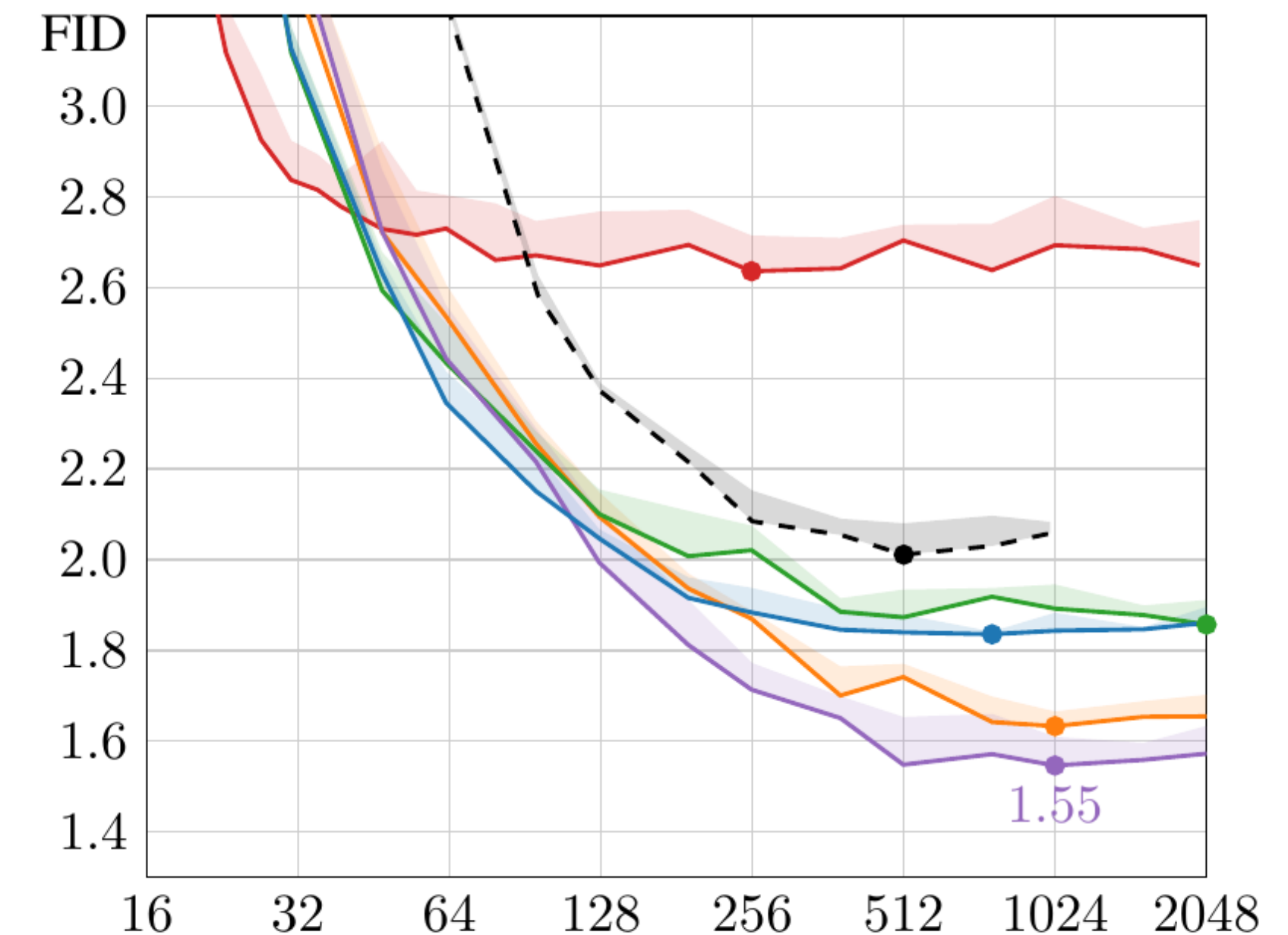
Same weights, **better sampler**: better images, fewer steps



(c) Class-cond. ImageNet-64, DDIM

Karras et al., 2022 · Fig. 2(c)

new sampler (green, red): same quality in far fewer network calls



(c) Class-cond. ImageNet-64

Karras et al., 2022 · Fig. 4(c)

purple: a tuned stochastic sampler, lower still

→ A better sampler alone improves an already-trained model: training and sampling are separate choices.

— CHAPTER 3 · 2021 → 2023

Text-to-image goes mainstream: type a sentence, get a picture

GLIDE

DALL·E 2

Imagen

Latent Diffusion

Stable Diffusion

DALL·E 3

Background: DALL·E 2 sample · Ramesh et al., 2022 · Fig. 1 (adapted)

03

Eight months: from a GLIDE paper to **open SD weights**



→ In eight months, text-to-image went from an OpenAI paper to weights anyone could download.

GLIDE: text-conditioned diffusion **beats DALL·E**



Samples: Nichol et al., 2021 · Fig. 1 (8 of 16, adapted)

87%

HUMANS PREFER GLIDE · PHOTOREALISM

69%

HUMANS PREFER GLIDE · CAPTION MATCH

vs DALL·E with CLIP reranking
3.5B vs 12B parameters

→ A diffusion model fed the caption directly already beat the 12B autoregressive DALL·E in human ratings.

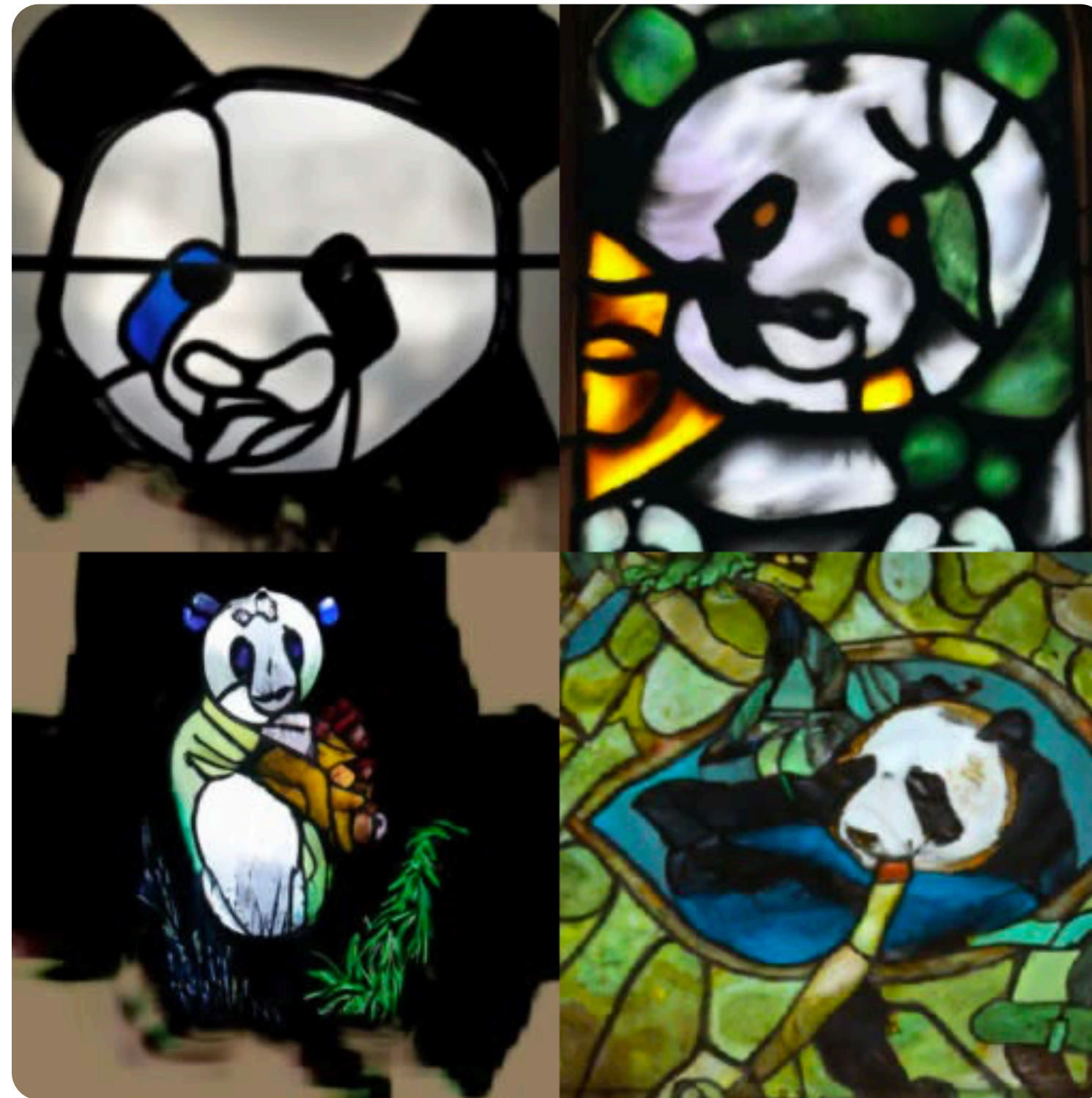
— GLIDE · "A STAINED GLASS WINDOW OF A PANDA EATING BAMBOO"

CLIP guidance wins on CLIP score; humans pick CFG



Nichol et al., 2021 · Fig. 17a (4 of 16, adapted)

DALL·E · CLIP-reranked



Nichol et al., 2021 · Fig. 17c (4 of 16, adapted)

GLIDE + CLIP guidance (higher CLIP score)

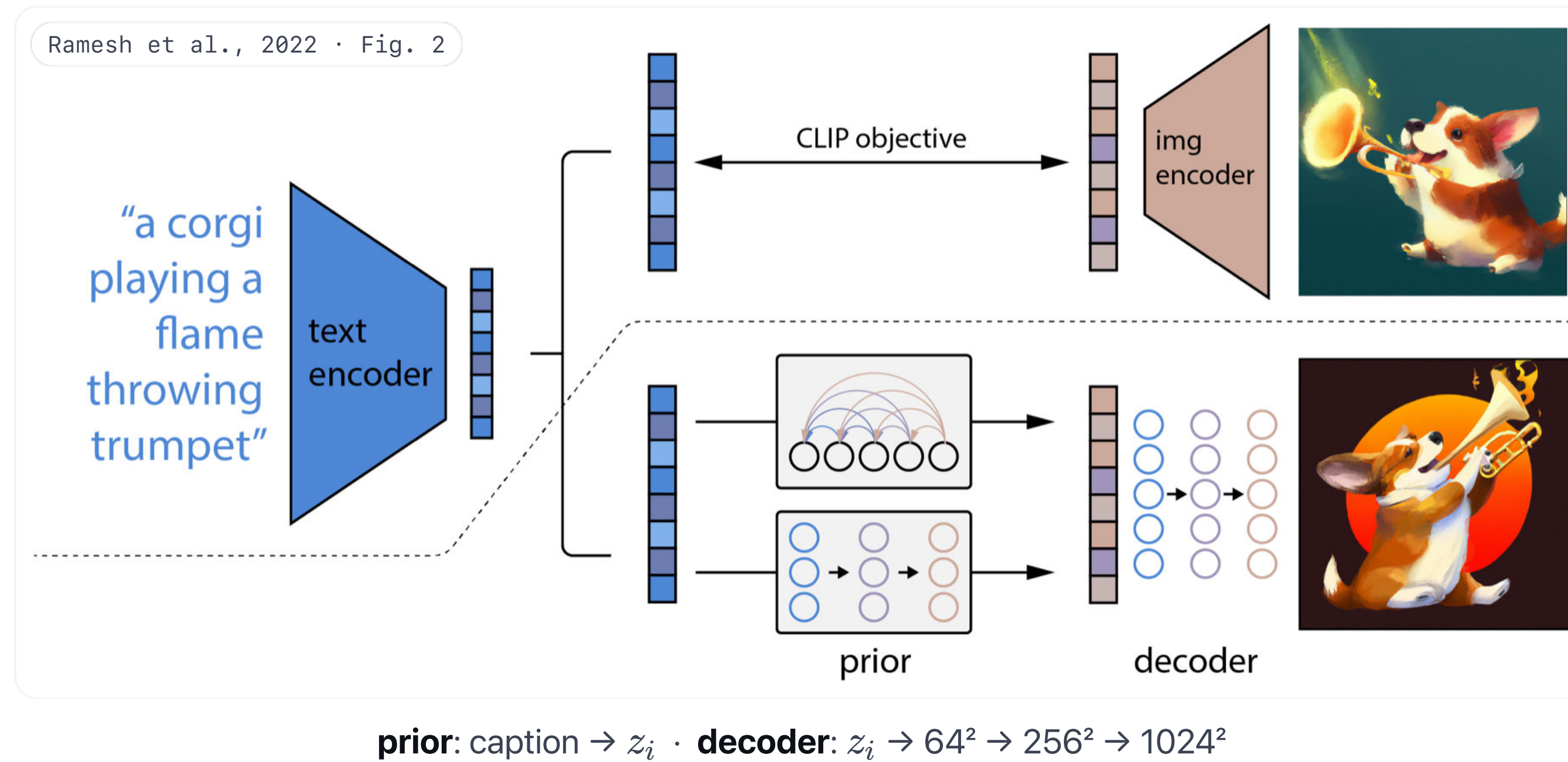


Nichol et al., 2021 · Fig. 17d (4 of 16, adapted)

GLIDE + CFG (humans prefer)

→ CLIP guidance scores higher on CLIP, yet humans prefer CFG; the authors suspect it fools the scorer.

DALL·E 2: predict a **CLIP image embedding**, then decode



TWO STAGES

$$P(x \mid y) = P(x \mid z_i, y) P(z_i \mid y)$$

exact, because the CLIP embedding z_i is a fixed function of the image x



Sample: Ramesh et al., 2022 · Fig. 1 (adapted)

→ Split the job: first guess what CLIP would "see" in the picture, then paint pixels that match it.

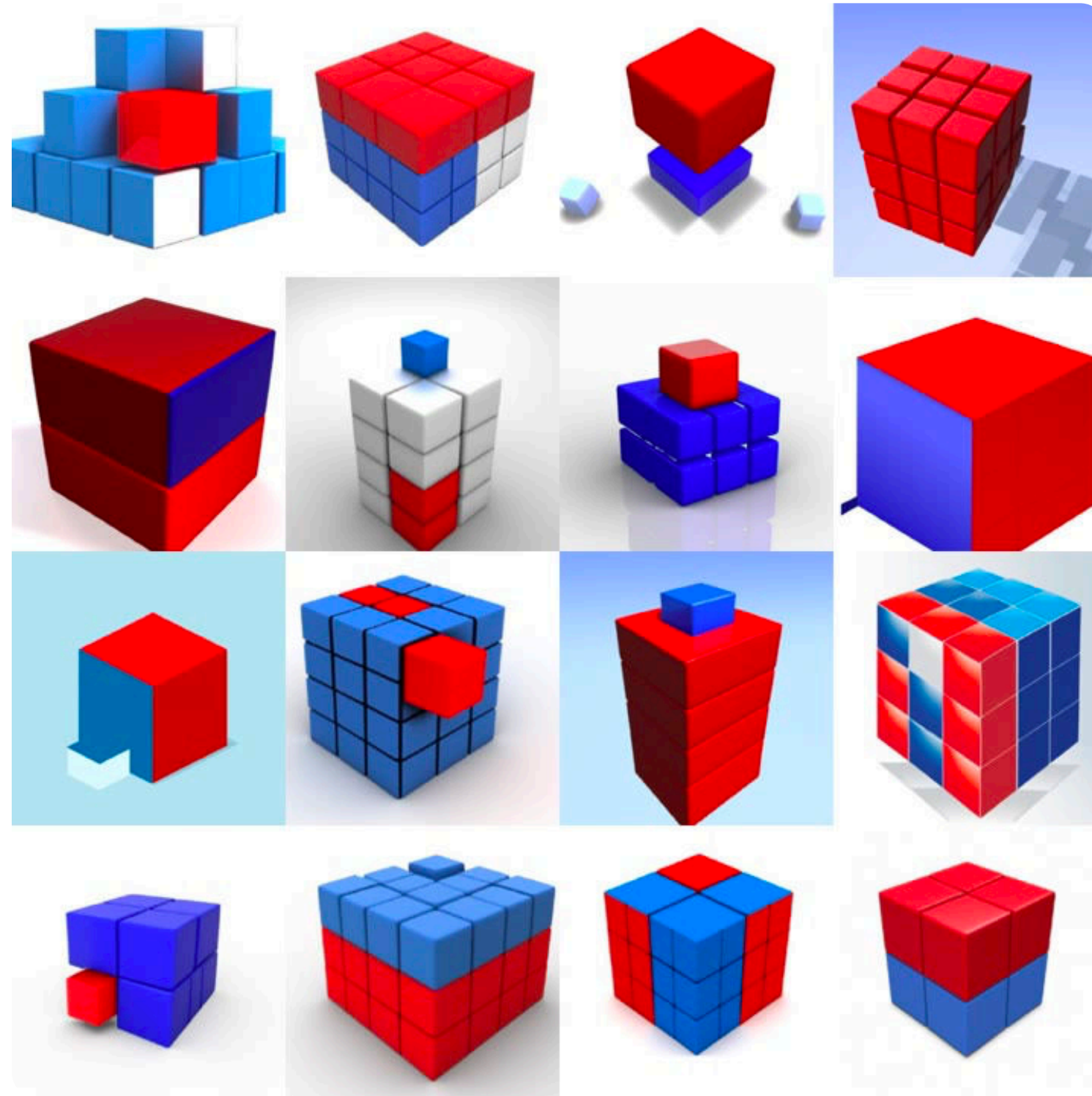
The embedding keeps the gist, **loses who is where**



Ramesh et al., 2022 · Fig. 3 (adapted)

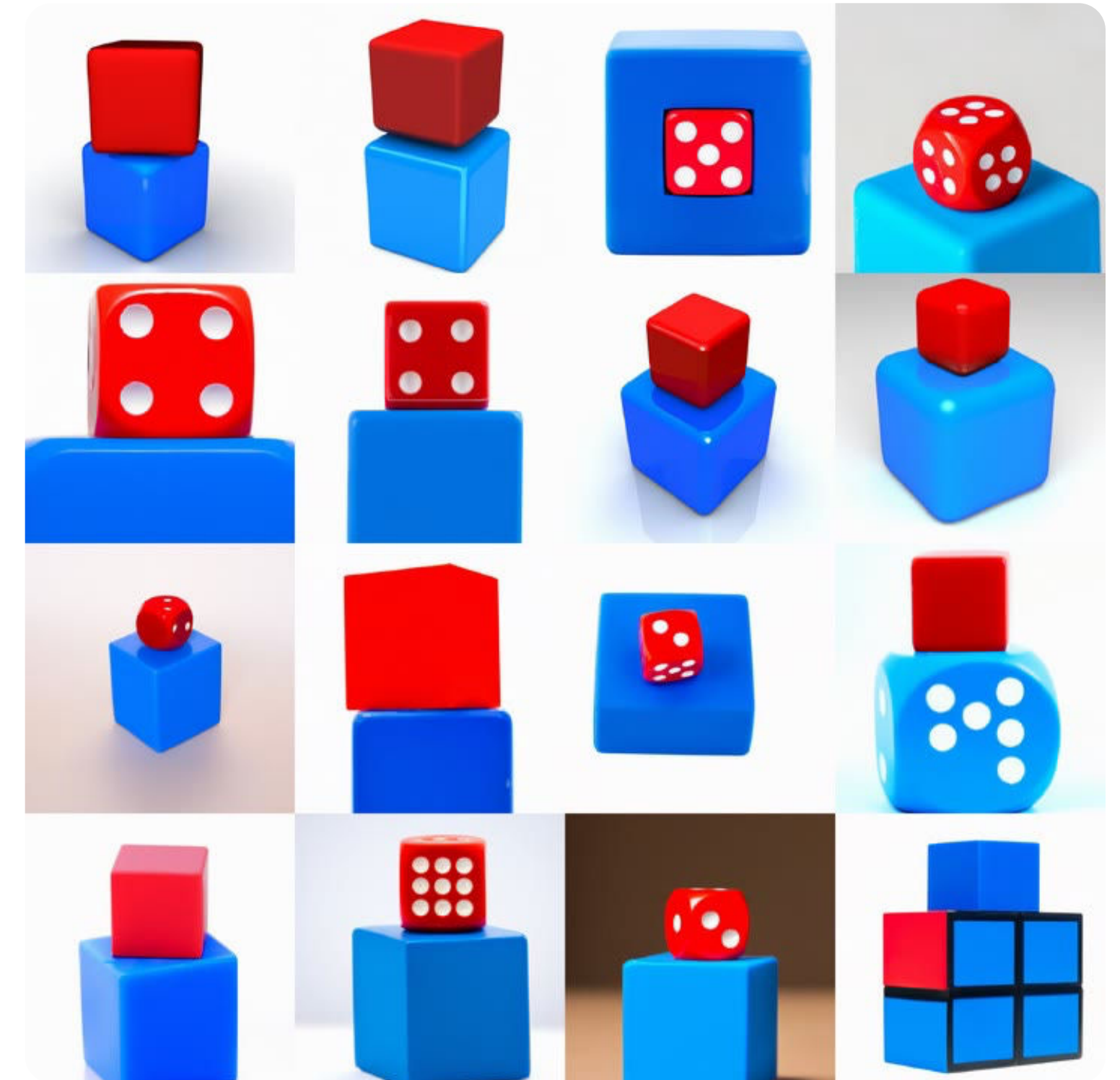


Input → **variations** of its embedding



Ramesh et al., 2022 · Fig. 14, unCLIP panel (adapted)

unCLIP · “a red cube on top of a blue cube”



Ramesh et al., 2022 · Fig. 14, GLIDE panel (adapted)

GLIDE · same prompt

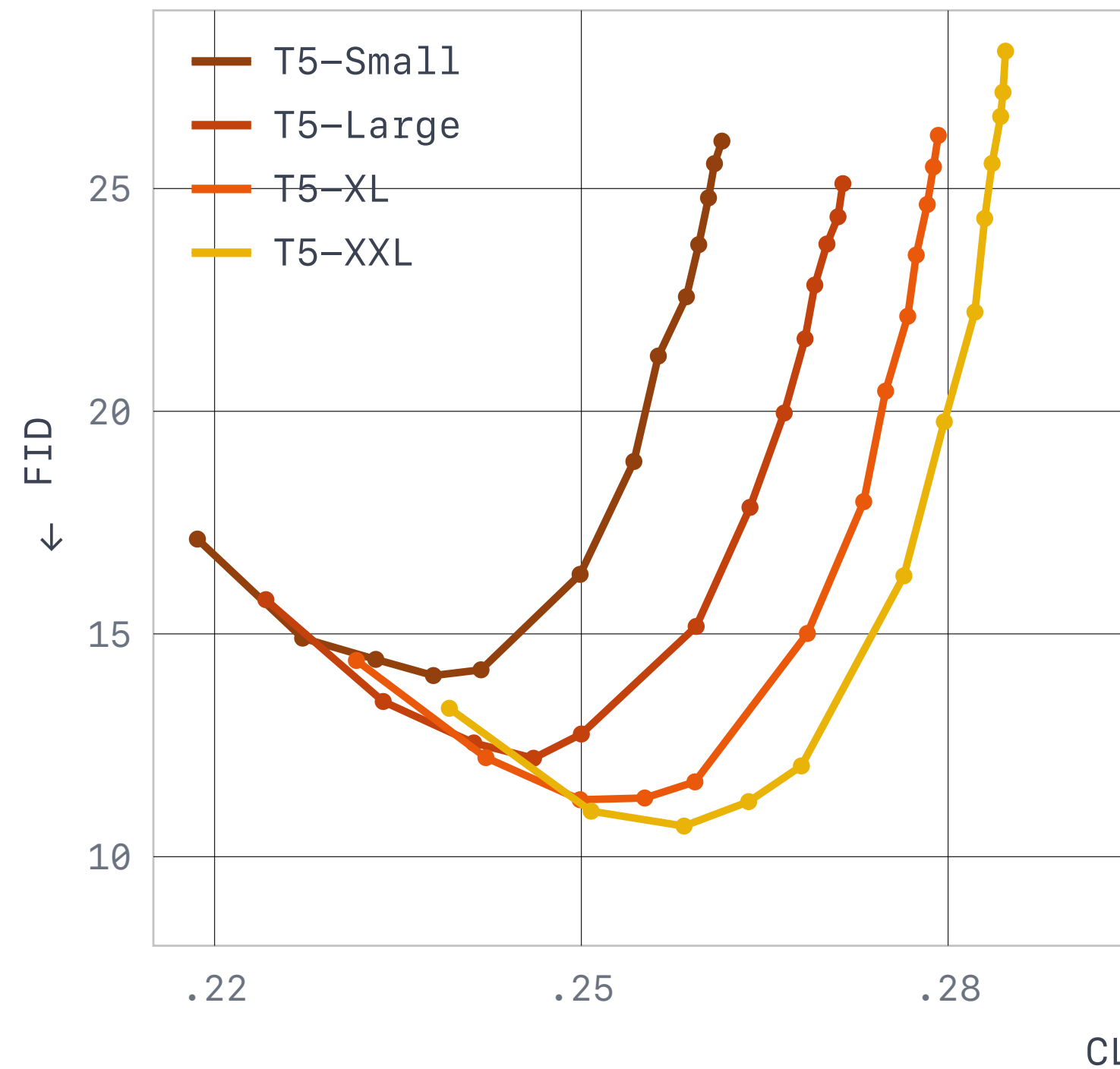
→ A CLIP vector captures what is in a picture, but only loosely which object has which color or where.

Imagen: a bigger **text encoder** beats a bigger U-Net

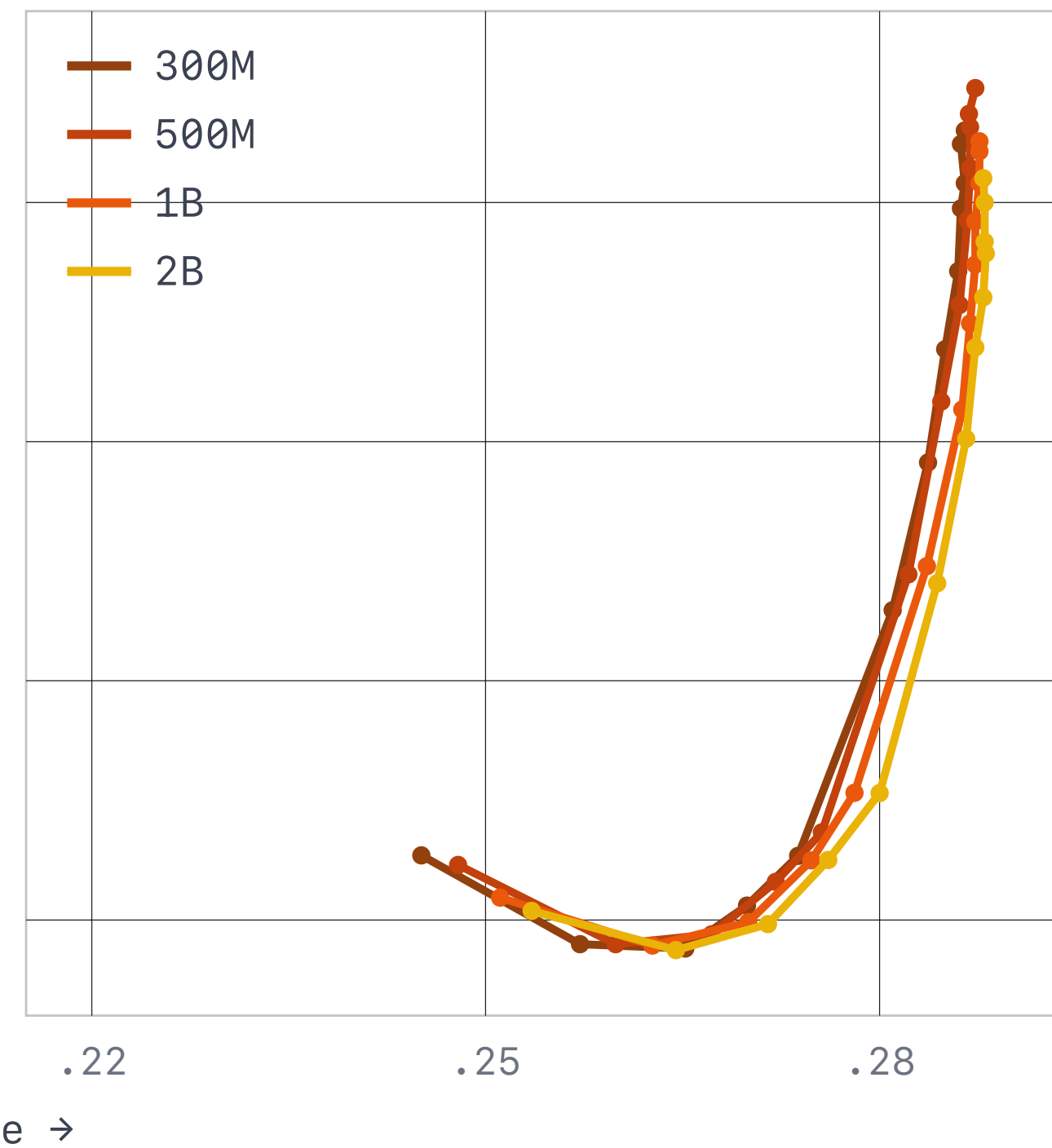
4.6B

FROZEN T5-XXL TEXT ENCODER

Bigger text encoder



Bigger U-Net



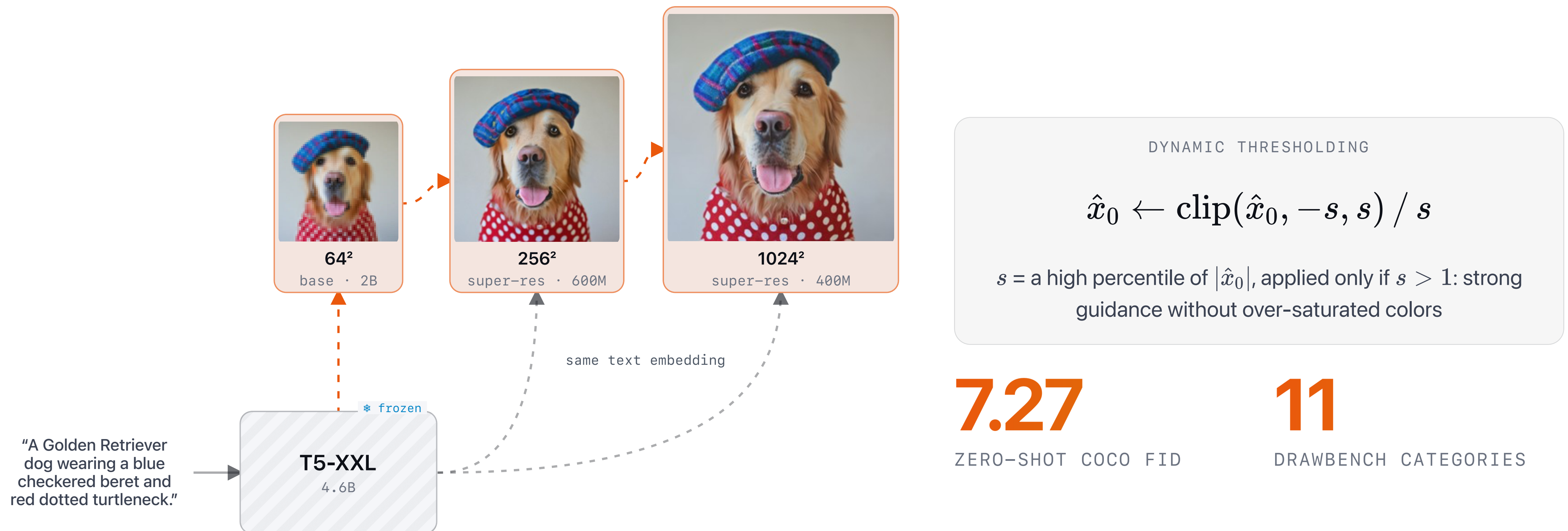
Data: Saharia et al., 2022 · Fig. 4a / 4b, replotted from the paper's source



Samples: Saharia et al., 2022 · Fig. 1 (adapted)

➔ Understanding the prompt was the bottleneck: a bigger frozen language model beat a bigger image model.

Imagen is a three-stage **pixel cascade** on frozen T5



After Saharia et al., 2022 · Fig. A.4 (adapted; sizes illustrative, sources ≤250 px)

→ Generate a tiny image, then upsample twice; every stage reads the same frozen text embedding.

Latent Diffusion: denoise a **small latent**, not pixels

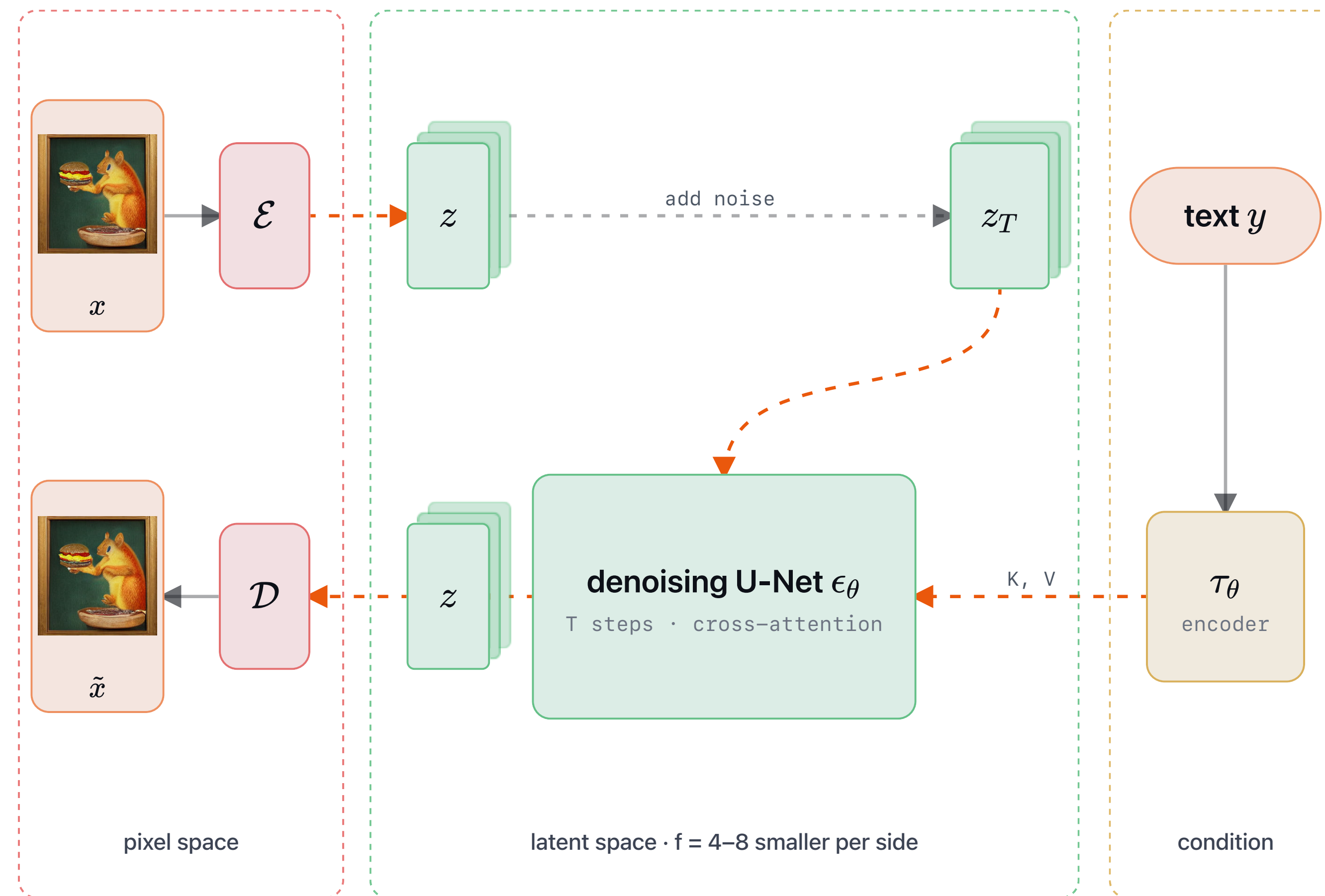


Image: Rombach et al., 2021 · Fig. 5 sample (adapted)

after Rombach et al., 2021, Fig. 3

LDM OBJECTIVE

$$\mathcal{L} = \mathbb{E} \left\| \epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y)) \right\|^2$$

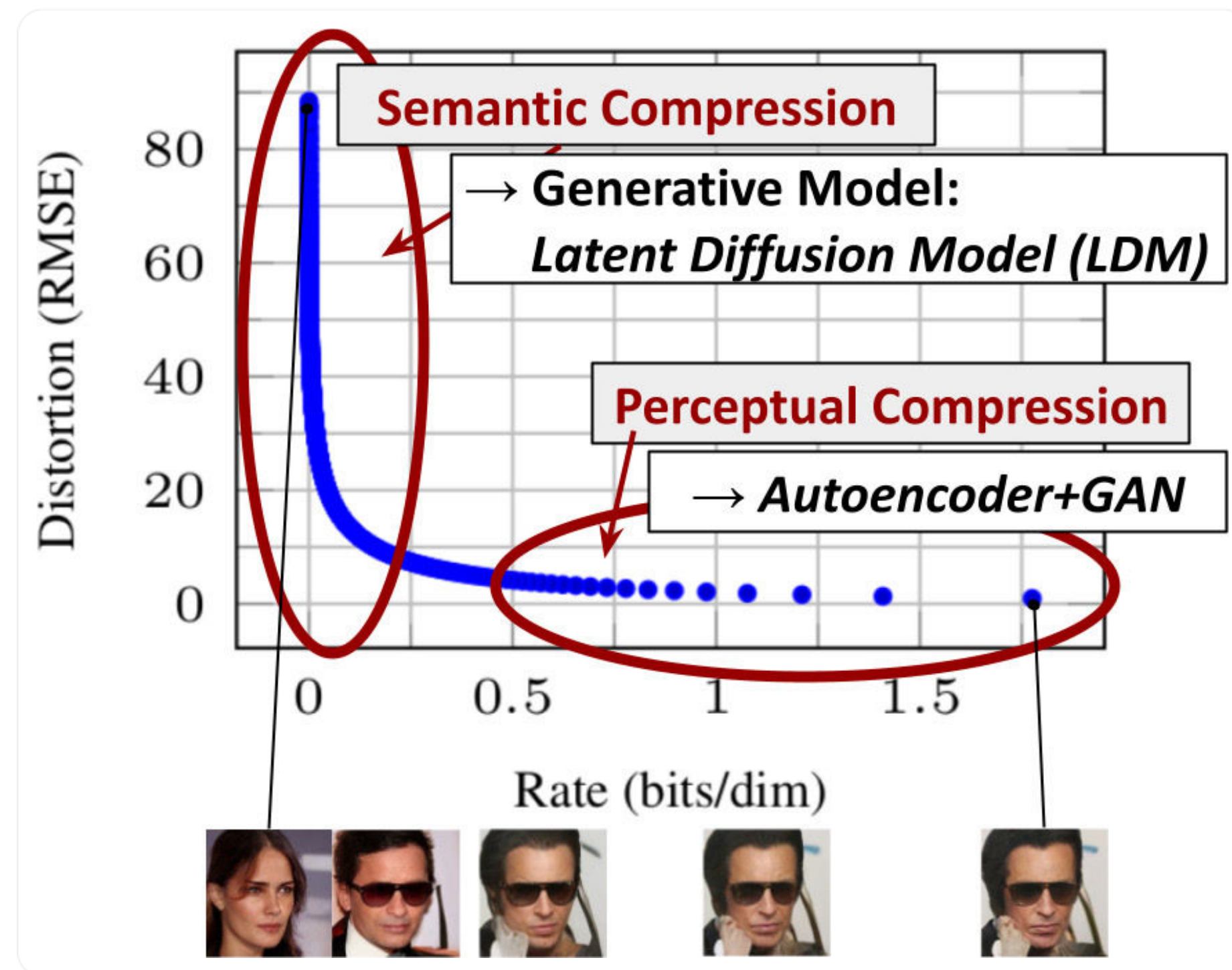
the primer's noise-prediction loss, applied to $z = \mathcal{E}(x)$ instead of pixels



Samples: Rombach et al., 2021 · Fig. 5 (1.45B LDM, adapted)

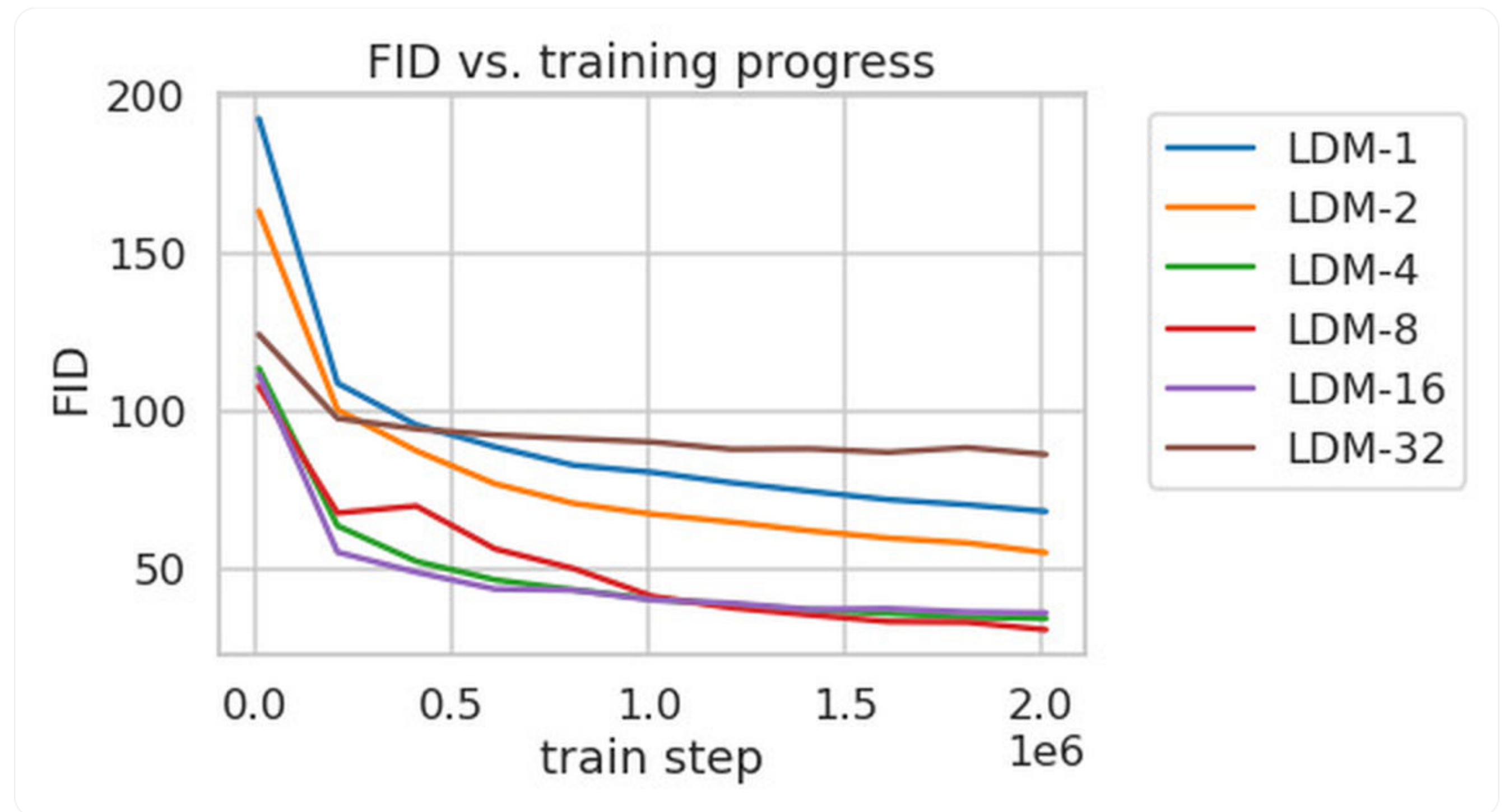
→ An autoencoder handles pixel detail, so diffusion only models a compact latent and gets far cheaper.

Compress away invisible bits, **diffuse the semantics**



Rombach et al., 2021 · Fig. 2 (data and images from Ho et al., 2020)

Few bits carry meaning; most are invisible detail

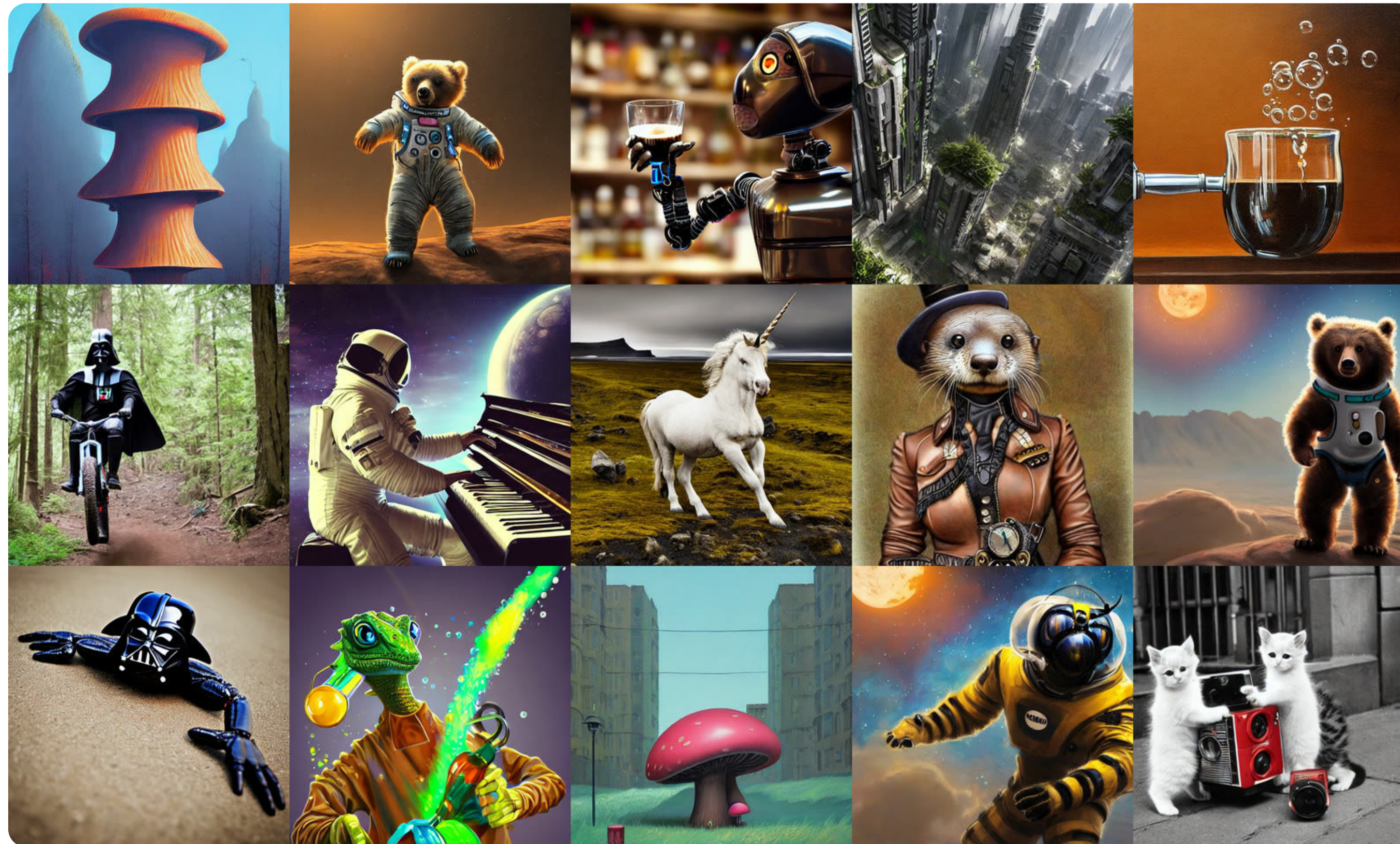


Rombach et al., 2021 · Fig. 6 (left panel, adapted)

Downsampling f: 1 is slow, **32** is lossy

→ Mild compression removes detail nobody sees; too little is slow, too much costs quality.

Stable Diffusion: **open weights** on a consumer GPU



Samples: CompVis, Stable Diffusion v1 (2022) · sample sheets (adapted)

860M **123M**

LATENT U-NET, $F = 8$ FROZEN CLIP TEXT

LAION-5B subsets · 512^2 · ~150k A100 GPU-hours

Diffusers

AUTOMATIC1111

ComfyUI

DreamBooth

LoRA

ControlNet

→ Same latent recipe, but public weights: anyone could run it, fine-tune it and build tools on top.

DALL·E 3: the bottleneck was **the captions**

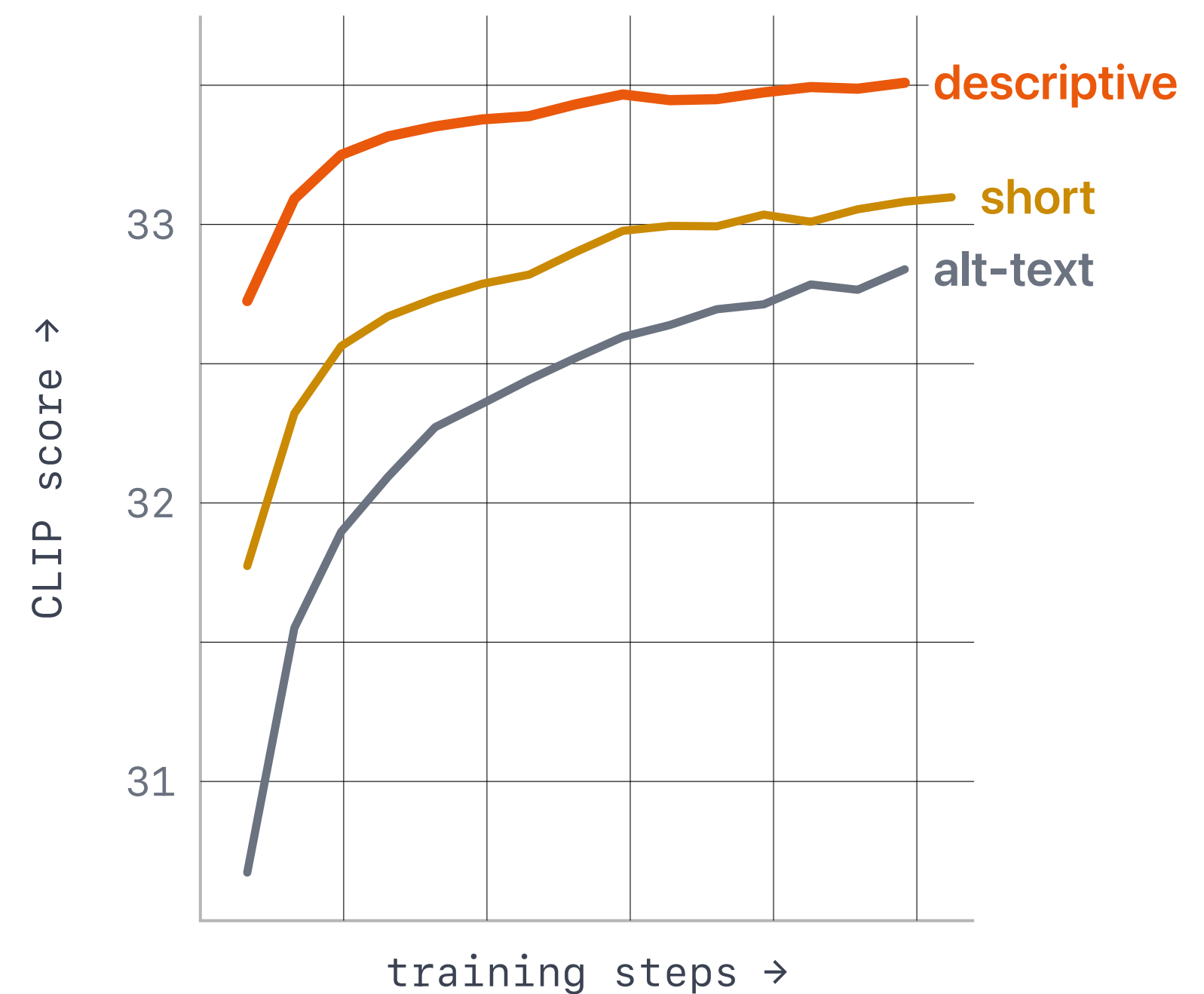


WEB ALT-TEXT
is he finished...just about!

SHORT SYNTHETIC
a quilt with an iron on it.

DESCRIPTIVE SYNTHETIC · 95%
...patchwork design with pastel-colored
strips of fabric and floral patterns...

Betker et al., 2023 · Fig. 3 (adapted; captions re-typeset)

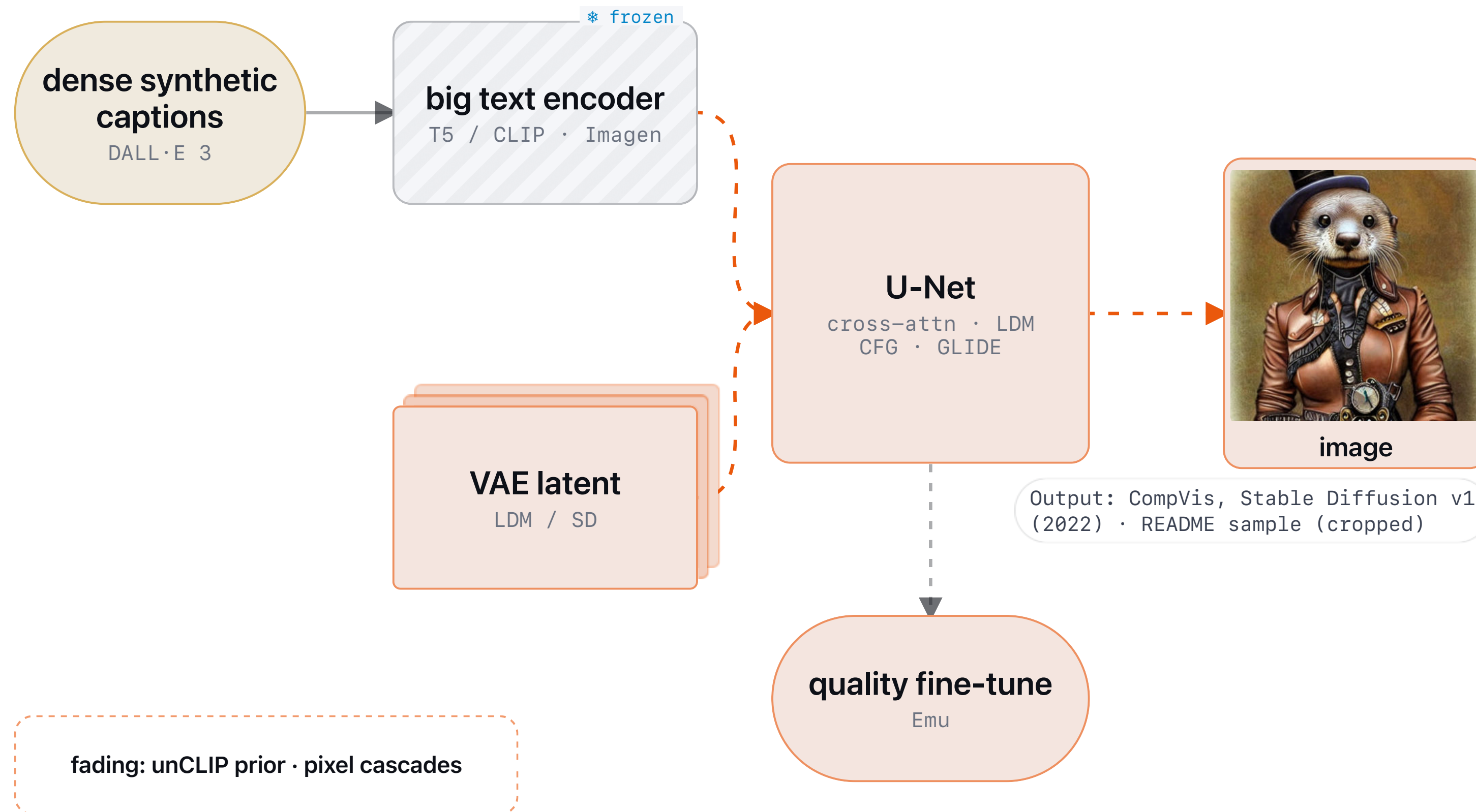


Data: Betker et al., 2023 · Fig. 4 (right), replotted

Descriptive > short > alt-text · ablations

→ Detailed training captions teach the model to follow long prompts.

By 2023 the recipe had **converged**



BUILT LARGELY FROM THESE PARTS

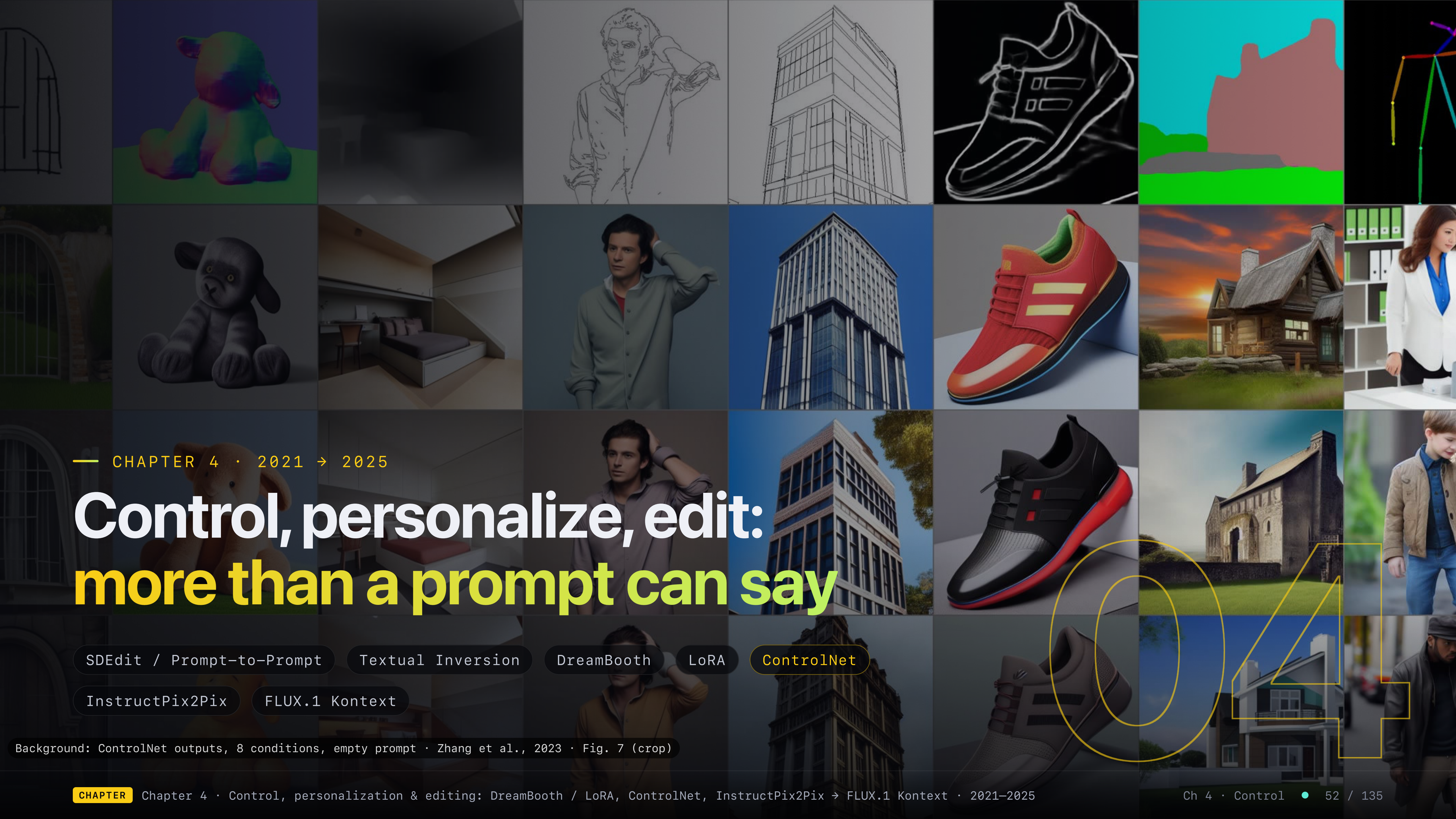
SDXL · 2023.07

Emu · 2023.09

DALL·E 3 · 2023.09 partly disclosed

Next: control and personalization on top of open SD (chapter 4)

→ By late 2023 most new systems used the same parts: a latent, a big text encoder, CFG and rich captions.



— CHAPTER 4 · 2021 → 2025

Control, personalize, edit: more than a prompt can say

SDEdit / Prompt-to-Prompt

Textual Inversion

DreamBooth

LoRA

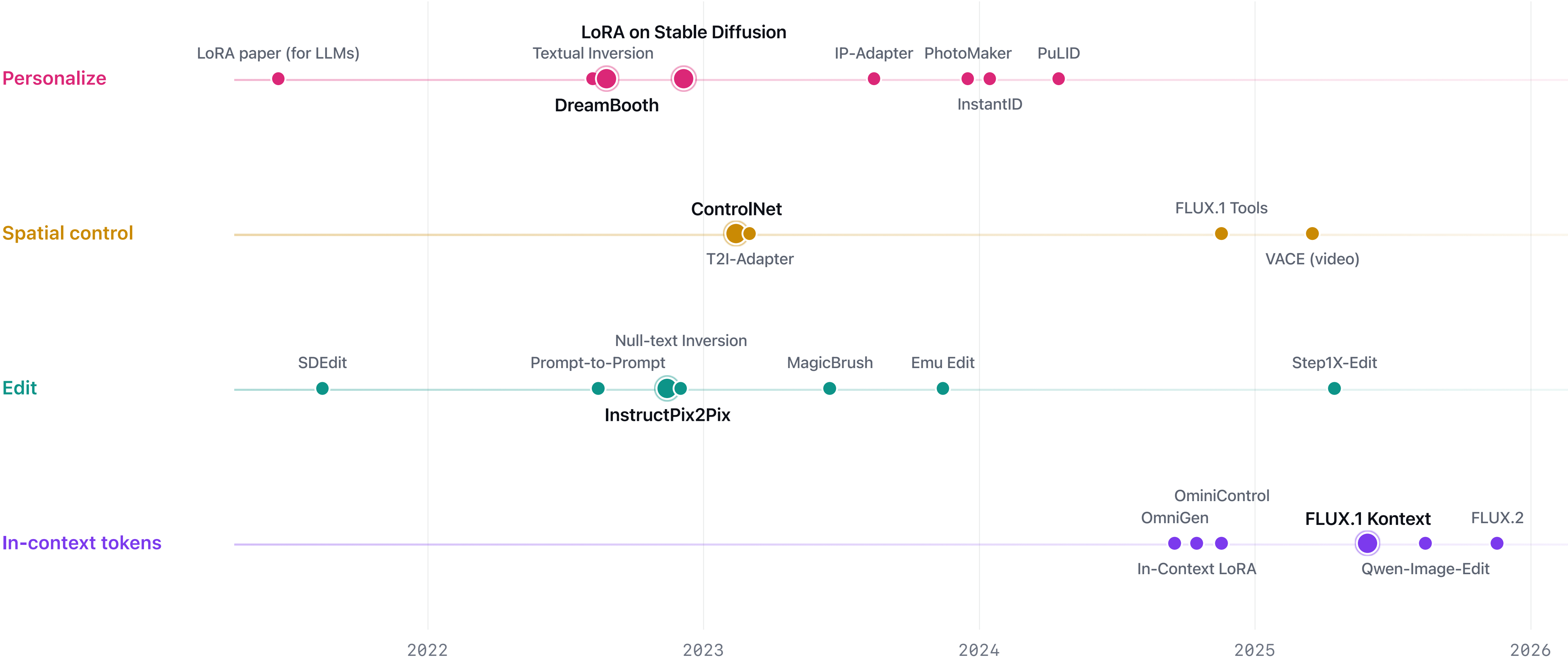
ControlNet

InstructPix2Pix

FLUX.1 Kontext

Background: ControlNet outputs, 8 conditions, empty prompt · Zhang et al., 2023 · Fig. 7 (crop)

Text can't say "my dog", "this pose", or "only here"



➔ Personalize a subject, fix the layout, edit one detail: each got its own fix, then one model did all three.

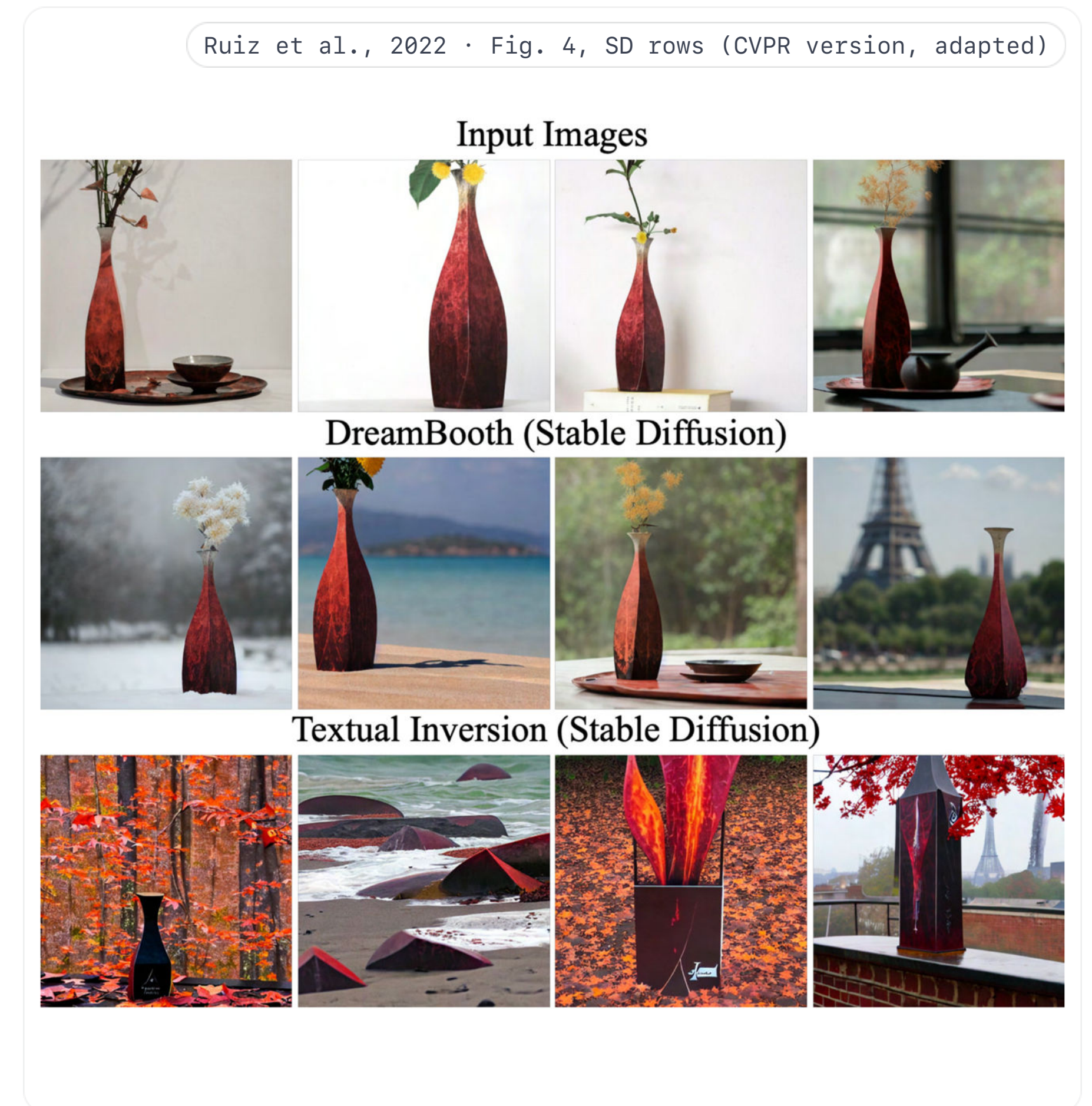
Teach the model **one specific dog** from 3–5 photos



PROMPT "a [V] dog" · PRIOR-PRESERVATION LOSS

$$\underbrace{\|\hat{x}_{\theta}(x_t, c_{[V]}) - x\|^2}_{\text{learn this dog}} + \lambda \underbrace{\|\hat{x}_{\theta}(x_{t'}^{pr}, c_{pr}) - x^{pr}\|^2}_{\text{replay the model's own "a dog" samples}}$$

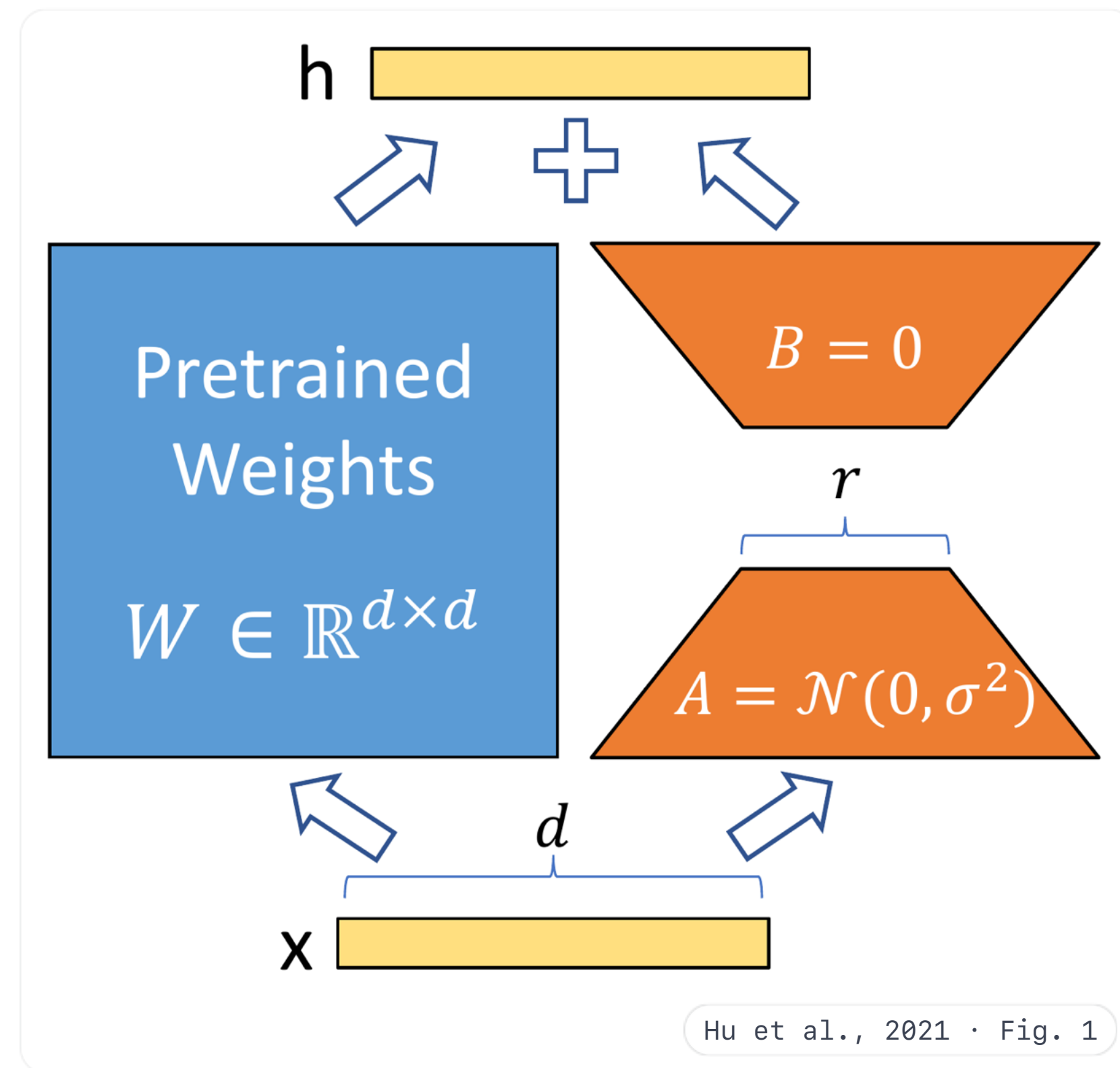
full fine-tune · ~1000 steps · ~5 min on one A100 (SD)



Textual Inversion learns one word vector; detail is lost

➔ Bind a rare word to your subject, fine-tune the model, and replay its own samples so "dog" keeps its meaning.

LoRA: ship a **few-MB patch**, not a GB checkpoint



FROZEN W_0 + LOW-RANK PATCH

$$W = W_0 + \frac{\alpha}{r} B A$$

$B = 0$: starts as the original model

1–6 MB

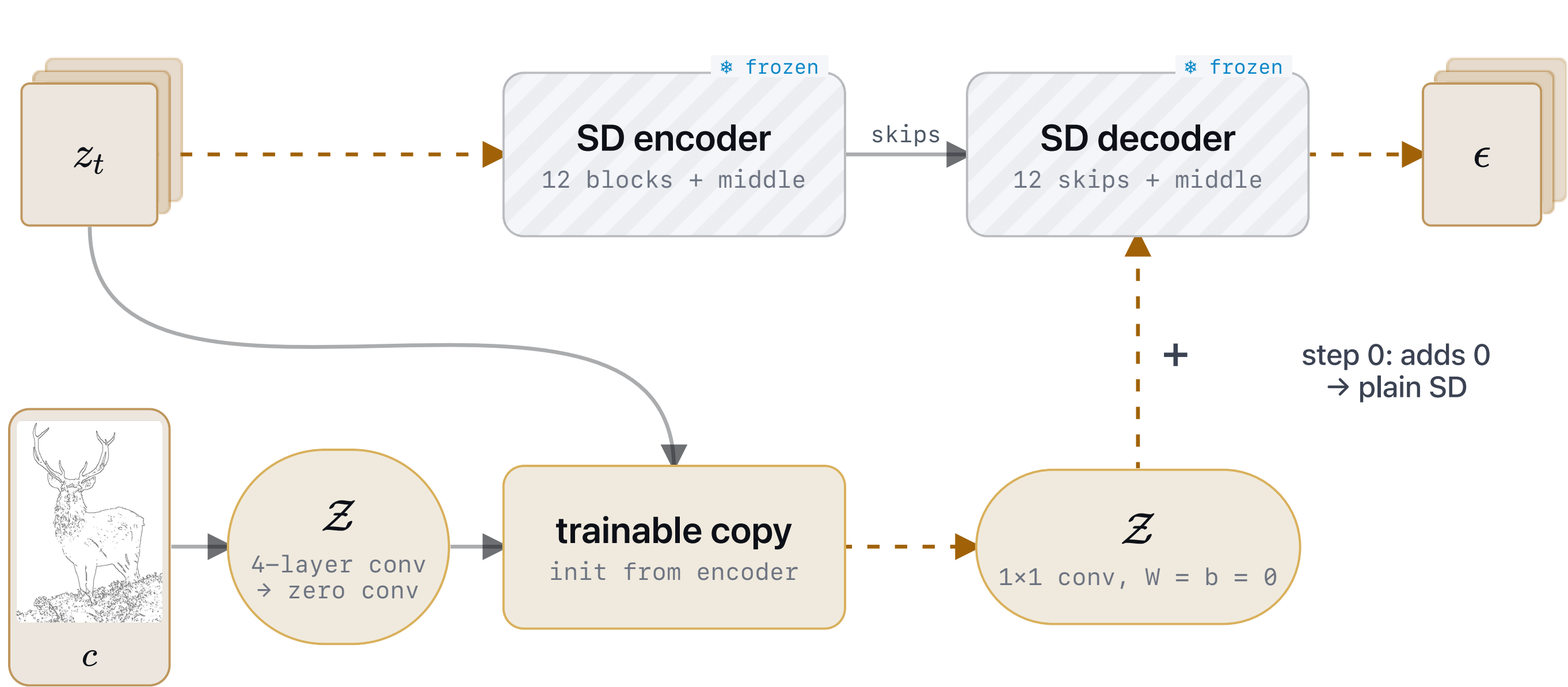
PER FILE · ~2× FASTER THAN FULL DREAMBOOTH (CLONEOFSIMO README)



LoRA scale s : 0 $\leftrightarrow$ 1 original $\leftrightarrow$ illustration

➔ Freeze the big weights, learn a tiny low-rank add-on; share it, stack it, dial its strength.

ControlNet: a trainable copy that **starts silent**



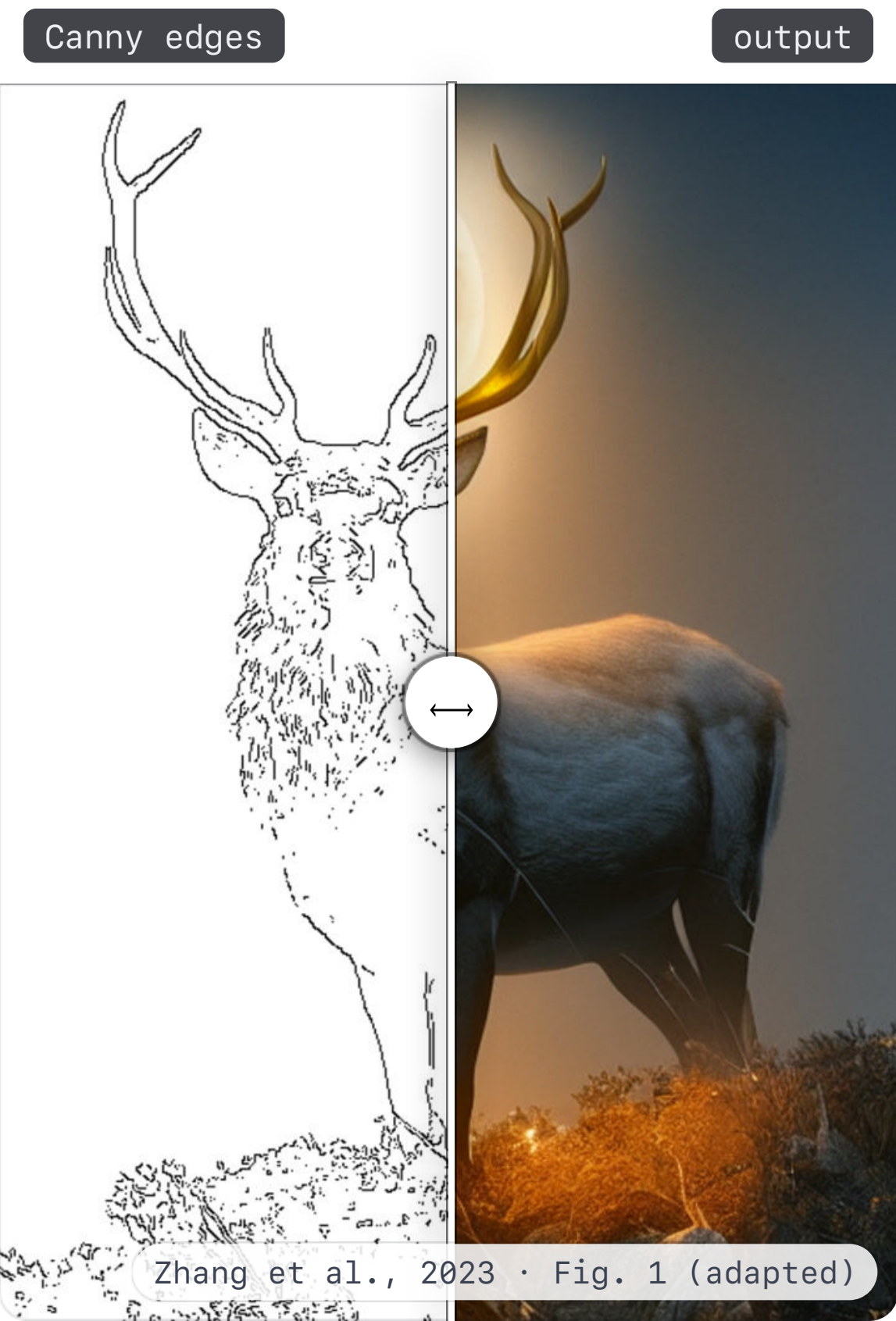
Edge map: Zhang et al., 2023 · Fig. 1 (adapted)

after Zhang et al., 2023, Fig. 3

ONE CONTROLNET BLOCK

$$y_c = \mathcal{F}(x; \Theta) + \mathcal{Z}(\mathcal{F}(x + \mathcal{Z}(c; \Theta_{z1}); \Theta_c); \Theta_{z2})$$

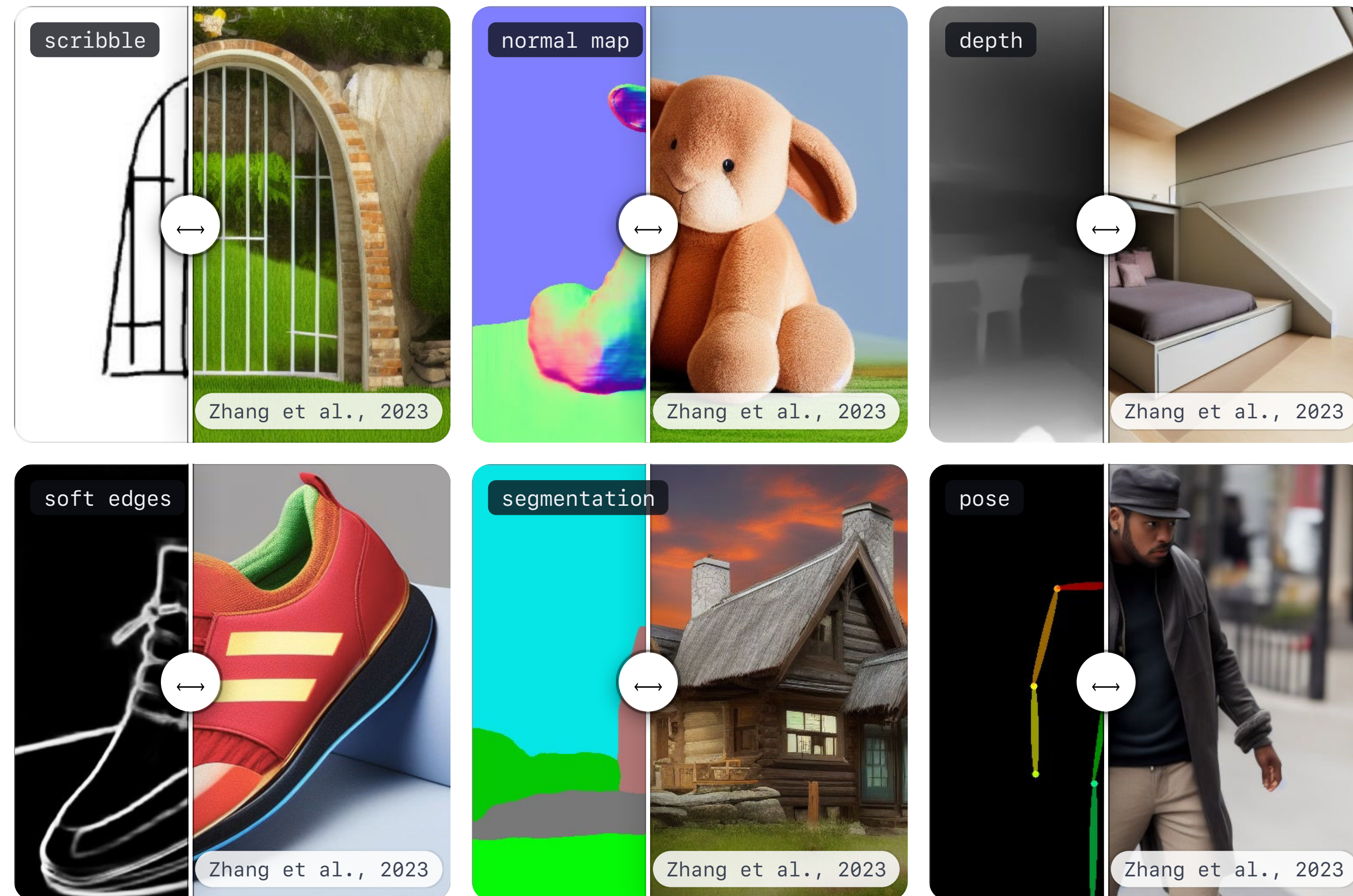
locked block + zero conv(copy(input + zero conv(condition)))



"masterpiece of fairy tale, giant deer, golden antlers"

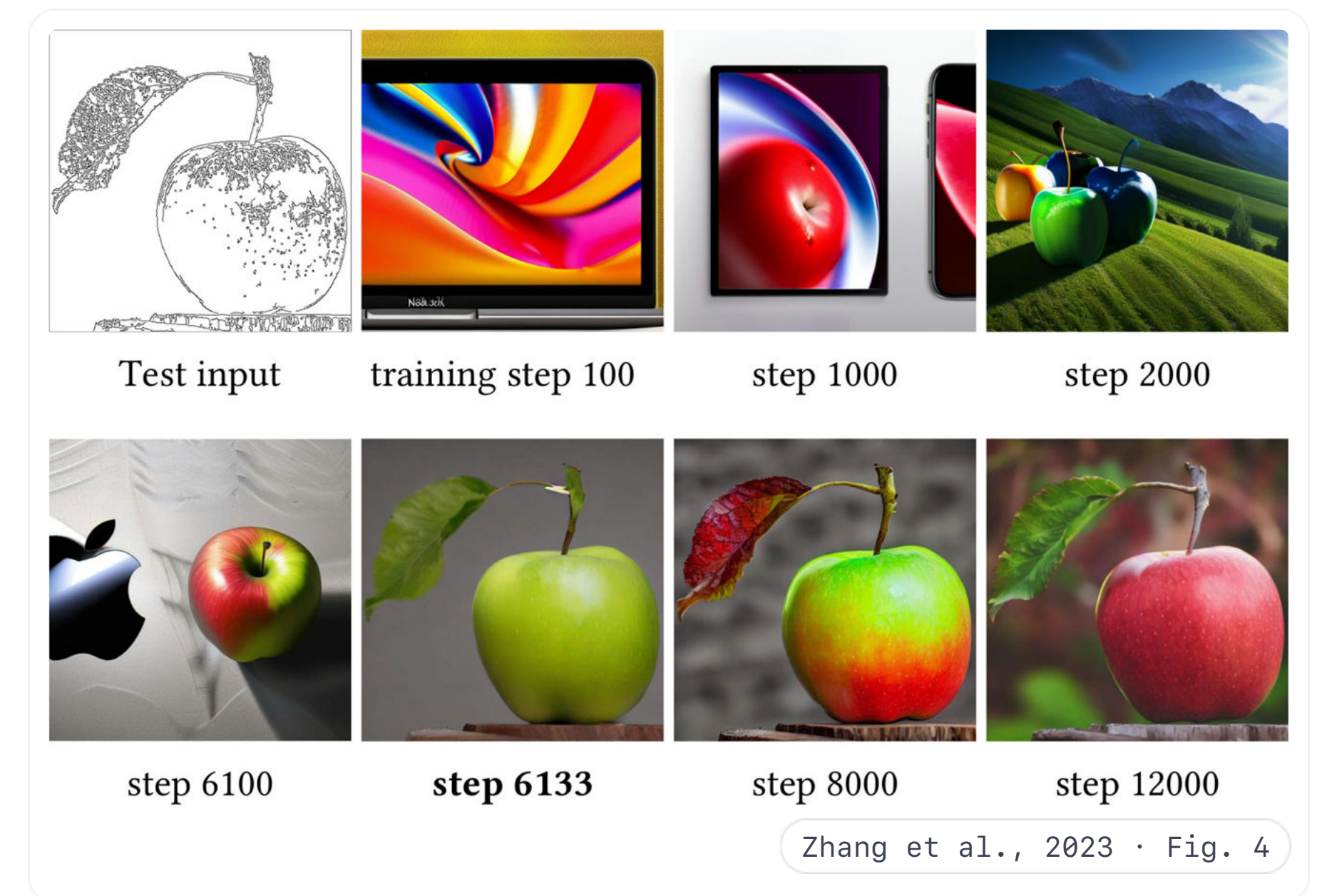
➔ Freeze Stable Diffusion; a copy reads the sketch or pose and joins through layers that start at zero.

Any spatial map works: edges, depth, pose, segments



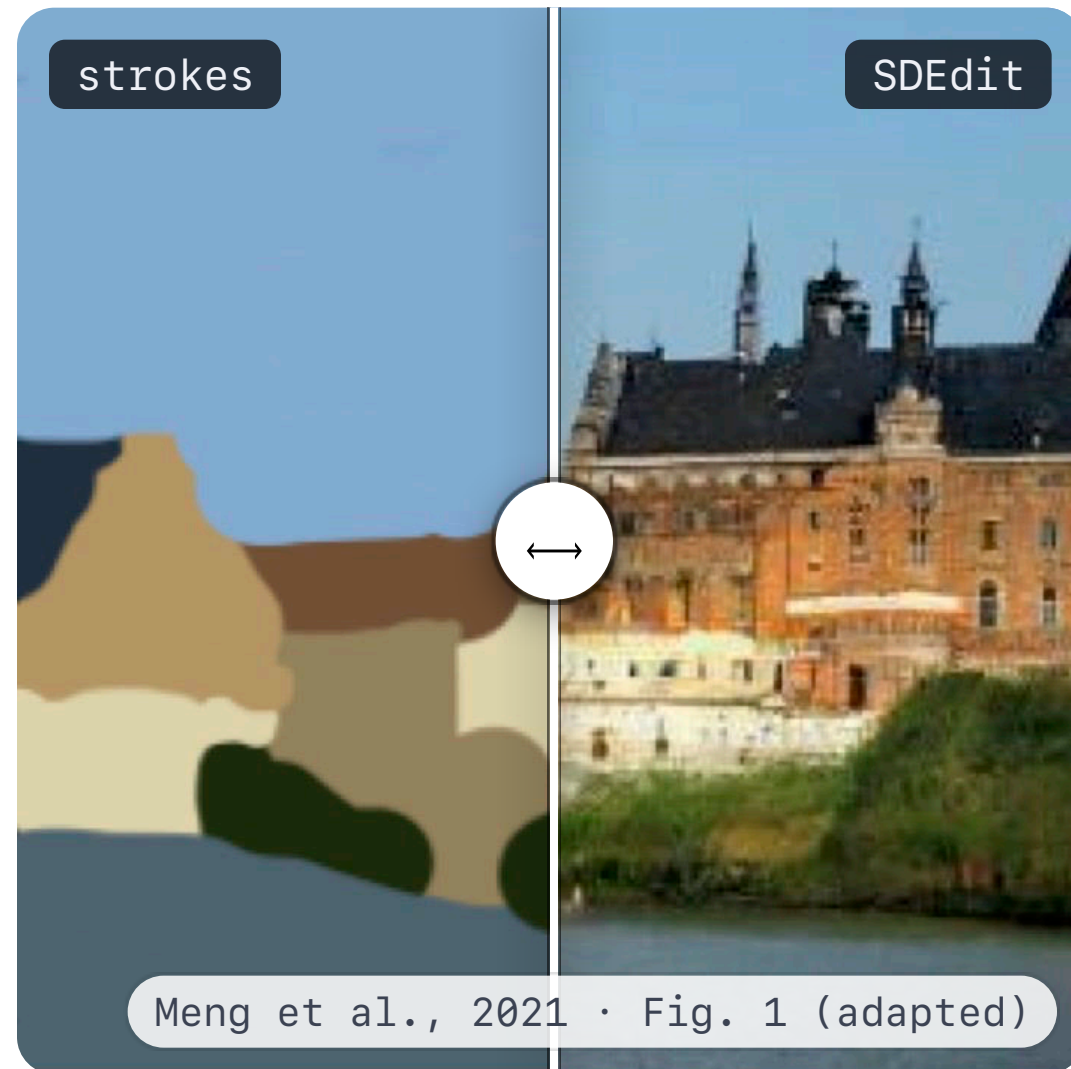
left of slider: condition · right: SD output, empty prompt · crops of Fig. 7

➔ Same frozen SD, one trainable encoder copy per condition: edges, depth, pose, segments, scribbles.

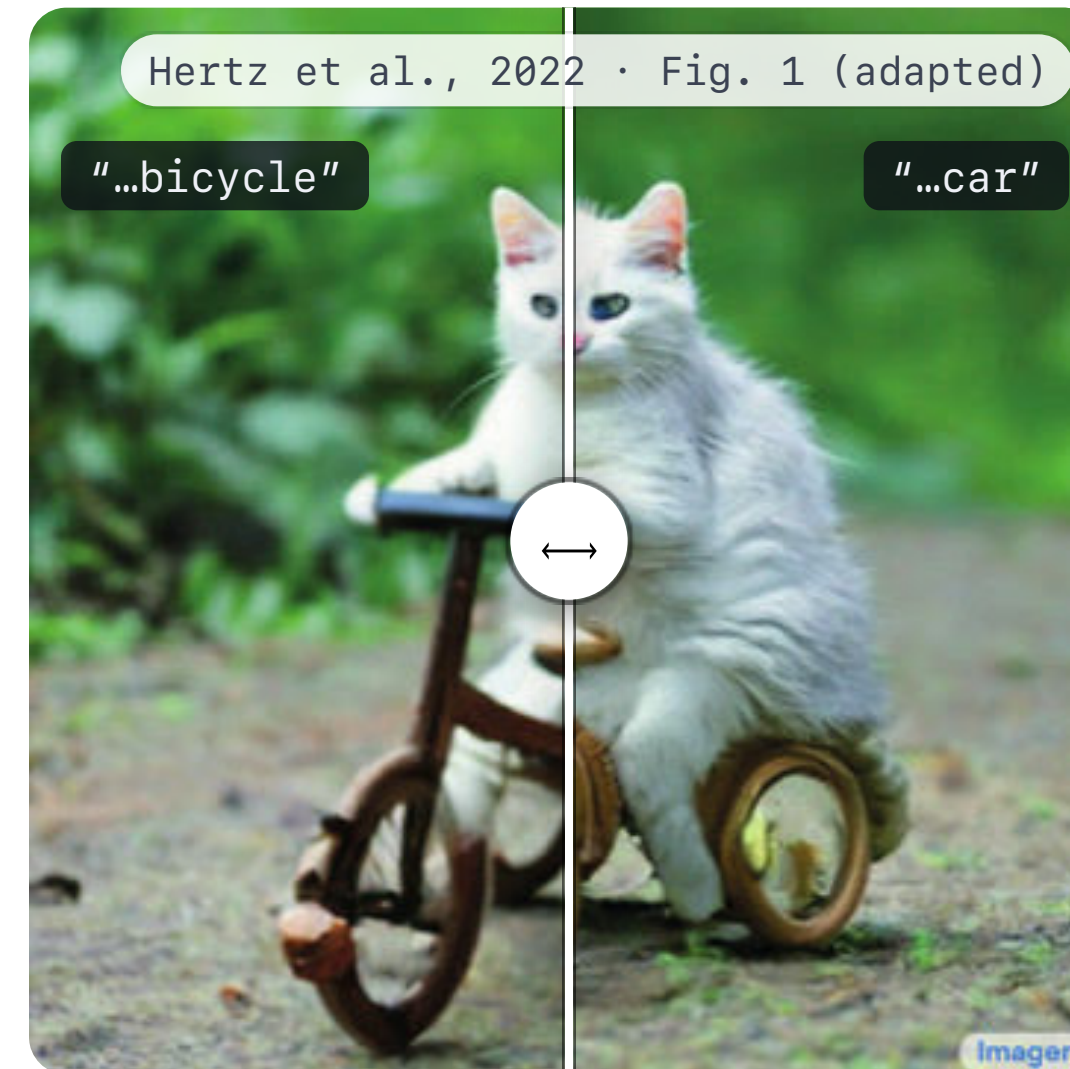


Sudden convergence: it snaps to the condition, usually within 10k steps

Editing without training: **noise it**, or **steer attention**



SDEdit: noise to t_0 , then denoise;
img2img "strength" = t_0 .



Prompt-to-Prompt: swap one word, keep the cross-attention maps: same layout.



$t_0 = 0.2$ · faithful

$t_0 = 0.5$ · balanced

$t_0 = 0.8$ · realistic

Meng et al., 2021 · Fig. 3(b) (adapted)

Catch: real photos need inversion first; Null-text Inversion takes minutes each.

→ Early editing hacked the sampler: partly re-noise a picture, or reuse which word was drawn where.

No editing dataset? **Generate one** with GPT-3 + SD

Training Data Generation

(a) Generate text edits:

Input Caption: *"photograph of a girl riding a horse"* → **GPT-3** → Instruction: *"have her ride a dragon"*
Edited Caption: *"photograph of a girl riding a dragon"*

(b) Generate paired images:

Input Caption: *"photograph of a girl riding a horse"*
Edited Caption: *"photograph of a girl riding a dragon"* → **Stable Diffusion + Prompt2Prompt** → 

(c) Generated training examples:



Brooks et al., 2022 · Fig. 2(a-c) (adapted)

700

HUMAN-WRITTEN EDITS



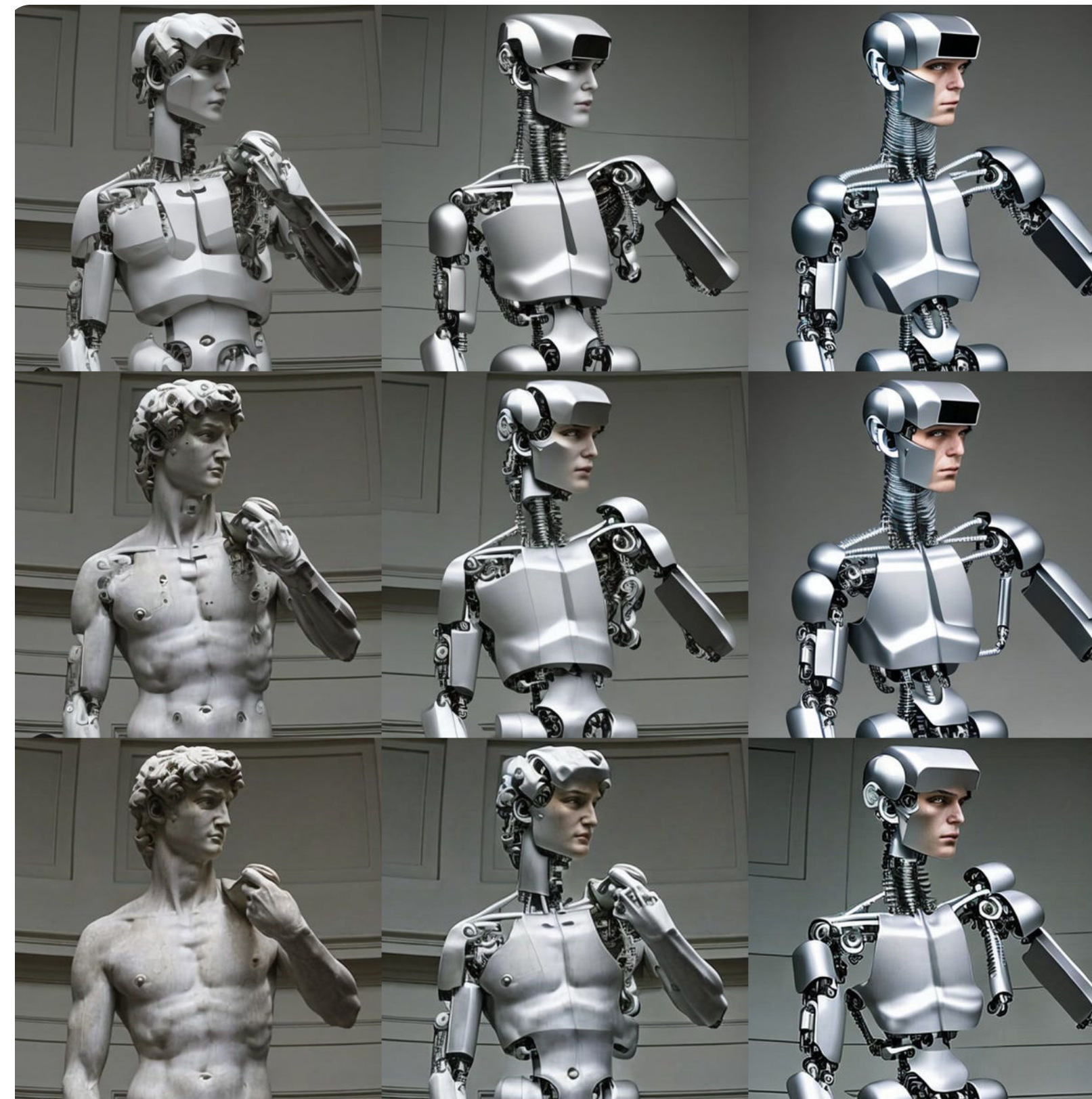
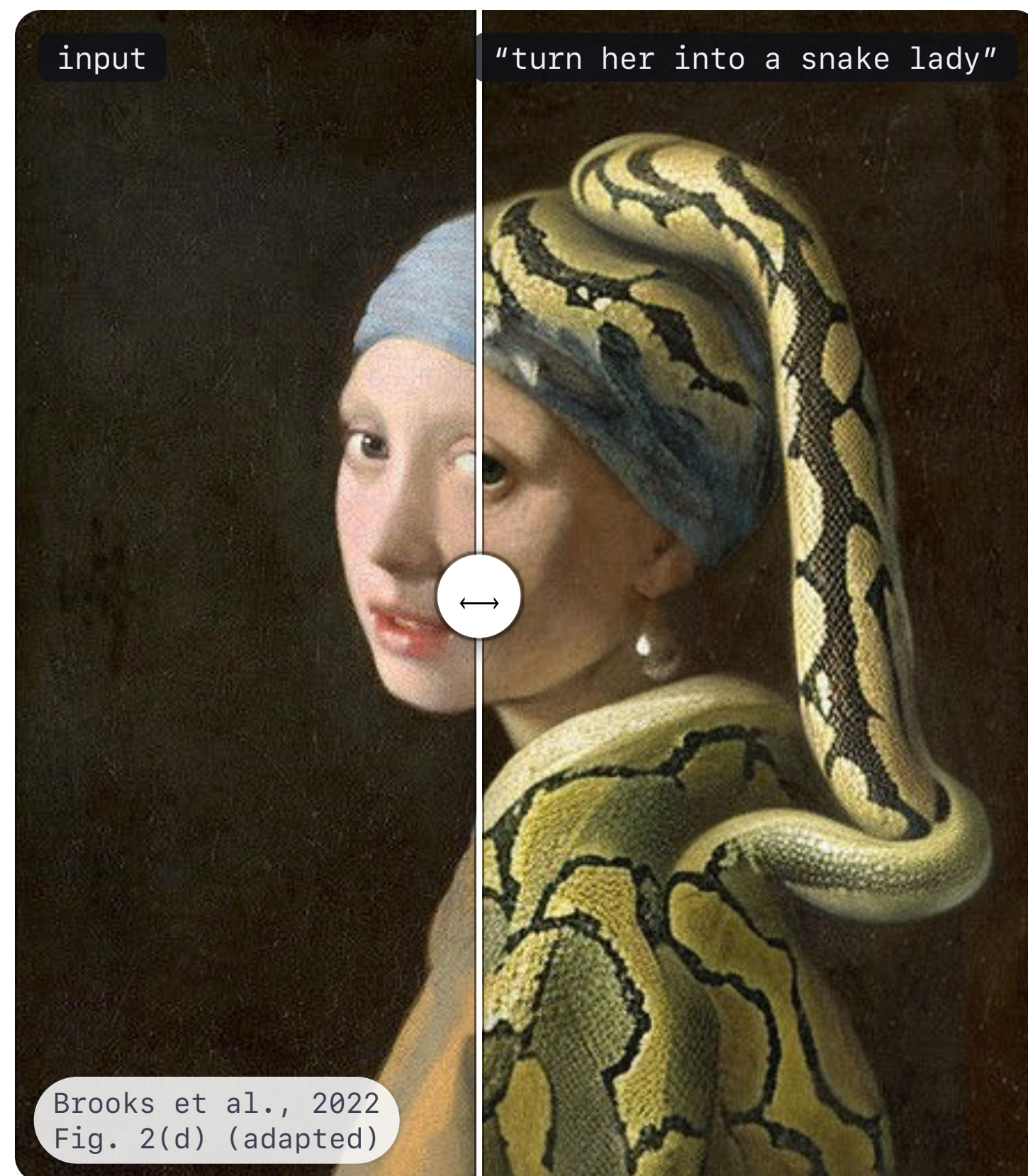
GPT-3 → SD + Prompt-to-Prompt
→ CLIP filter

454,445

SYNTHETIC TRAINING PAIRS

→ Teach GPT-3 to write edits, let Stable Diffusion paint before/after pairs, and get a huge synthetic dataset.

One instruction, one sampling run, **two guidance scales**



"Turn him into a cyborg!"
→ $s_T=3, 7.5, 15$ ↓ $s_I=1.0, 1.2, 1.6$

Brooks et al., 2022 · Fig. 4 (adapted)

TWO-CONDITION CFG

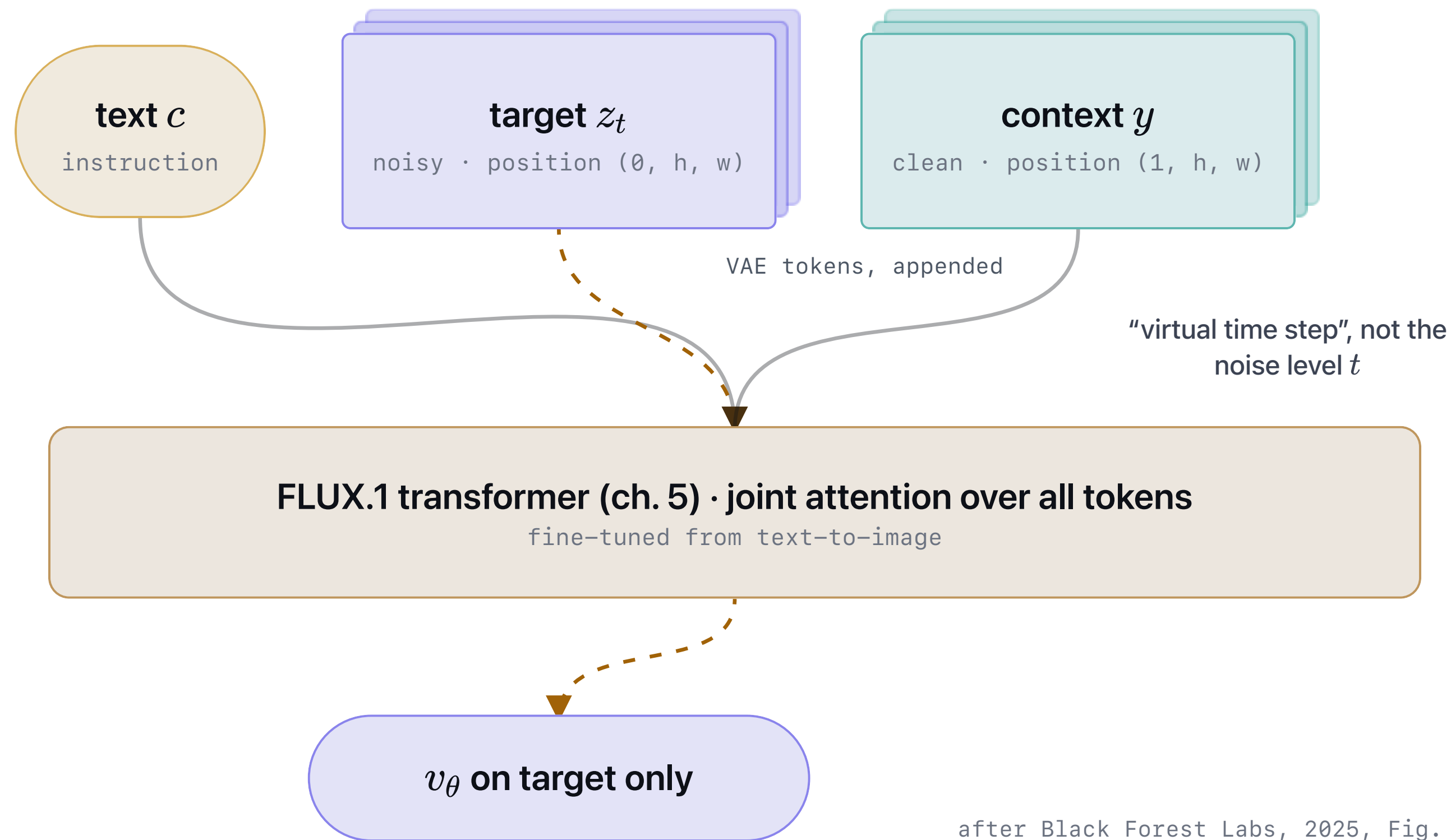
$$\begin{aligned}\tilde{\epsilon} = & \epsilon_{\emptyset\emptyset} \\ & + s_I (\epsilon_{I\emptyset} - \epsilon_{\emptyset\emptyset}) \\ & + s_T (\epsilon_{IT} - \epsilon_{I\emptyset})\end{aligned}$$

s_I : stay close to the image

s_T : follow the instruction

→ Trained only on synthetic pairs, it edits real photos from a plain sentence in seconds, no inversion.

Kontext: the reference image is **just more tokens**



Training: the usual denoising loss, on the target tokens only; drop the context ($y = \emptyset$) and it is plain text-to-image.

→ No side network: append the reference image's tokens to the sequence and let attention do the rest.



input



"remove the thing from her face"



"...taking a selfie in the streets of Freiburg"



"it's now snowing..."

Black Forest Labs, 2025 · Fig. 2 (adapted)

Same character, new scene: **edit after edit**



(a) context image



(b) "...sitting in a bar and enjoying a beer"



(c) "there are now two of these birds"



from (c): "watch them from behind"



from (c): grocery shopping



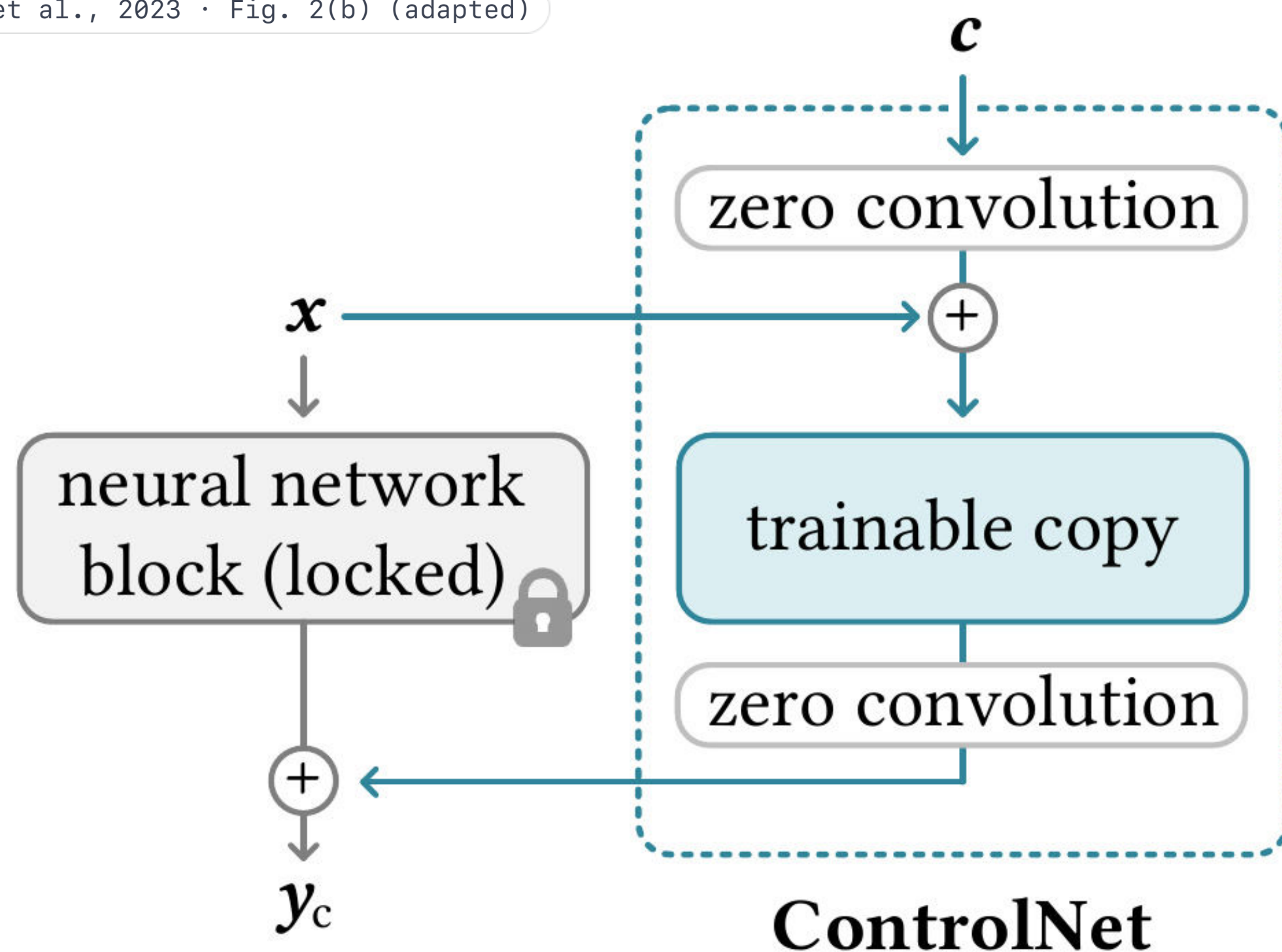
then: celebrating a successful launch

Black Forest Labs, 2025 · Fig. 1 (adapted)

➔ One reference image plus a sentence keeps a character consistent; no per-character fine-tuning needed.

Start from identity, then **let data do the work**

Zhang et al., 2023 · Fig. 2(b) (adapted)



STEP 0 = PRETRAINED MODEL

LORA

$$B = 0 \Rightarrow BA = 0$$

CONTROLNET

$$\mathcal{Z}(\cdot) = 0$$

INSTRUCTPIX2PIX

$$W_{\text{new channels}} = 0$$

U-Net era: side branches
ControlNet · IP-Adapter

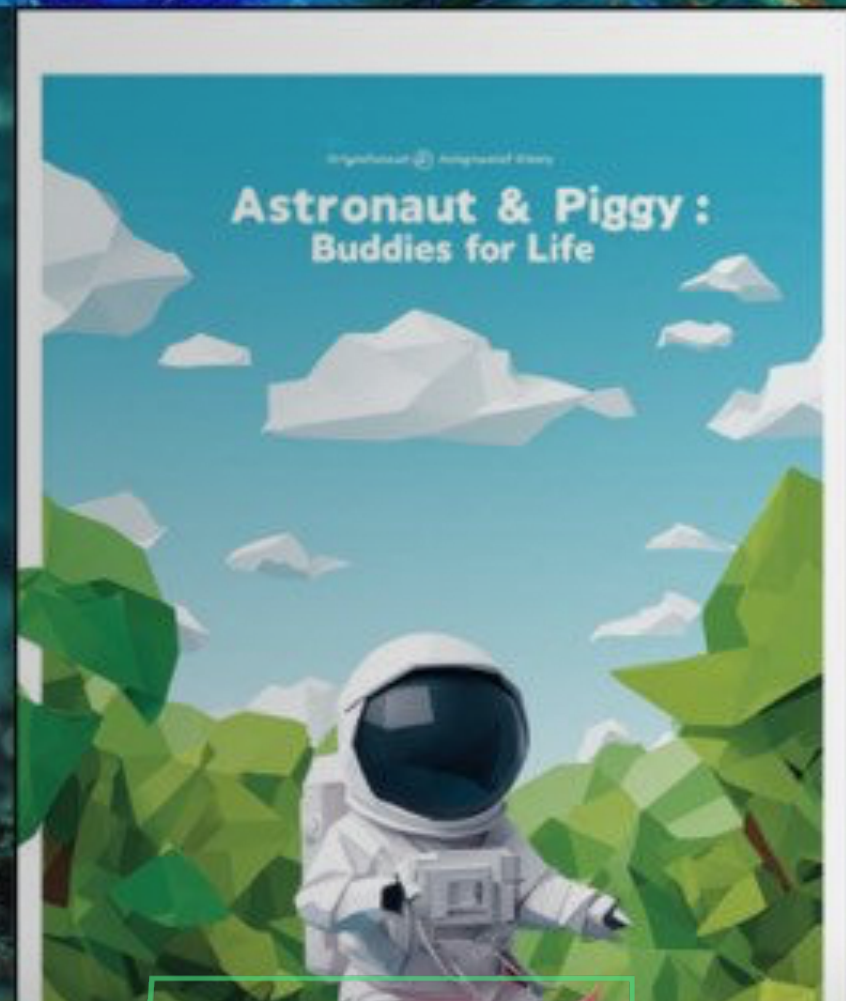
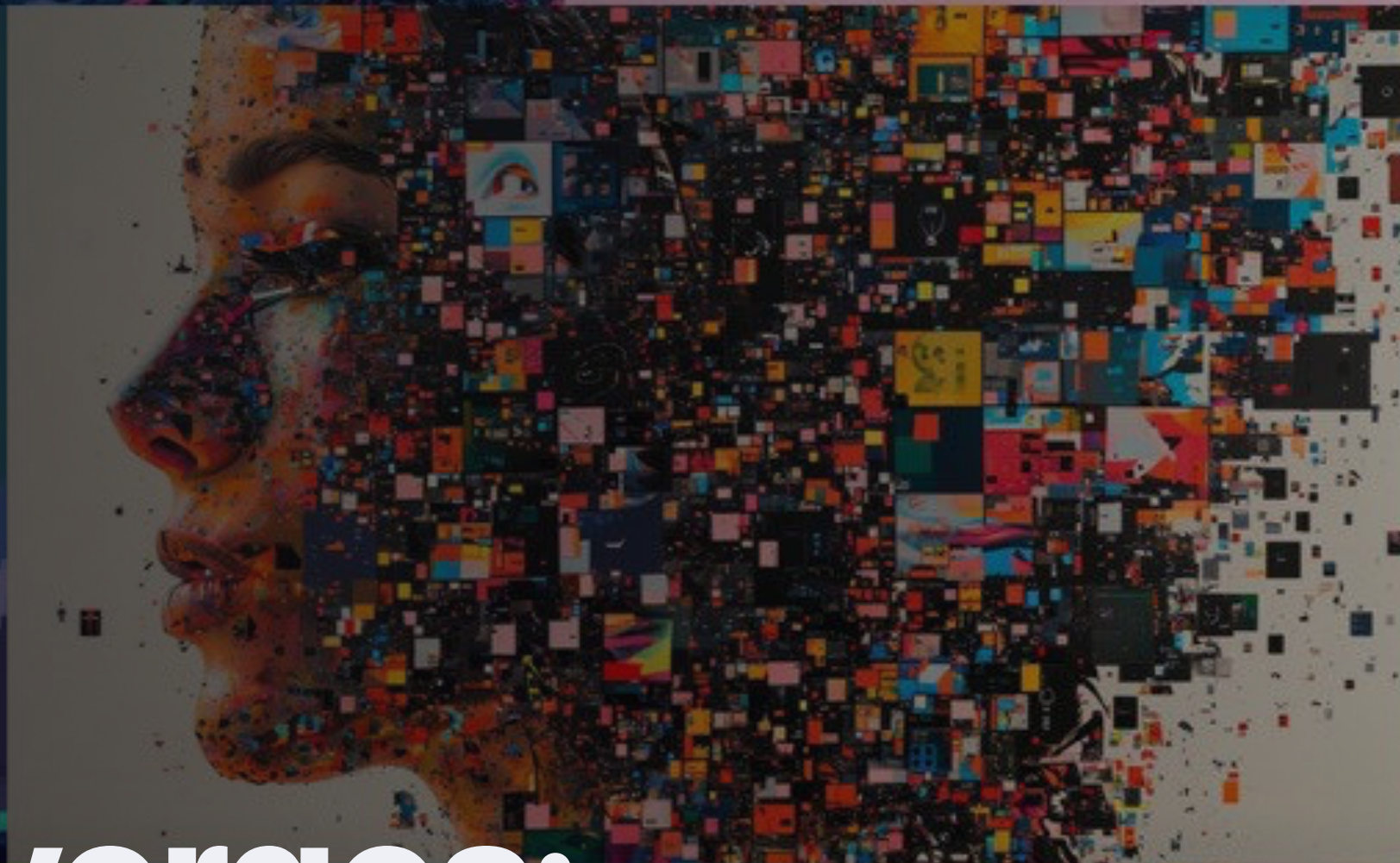
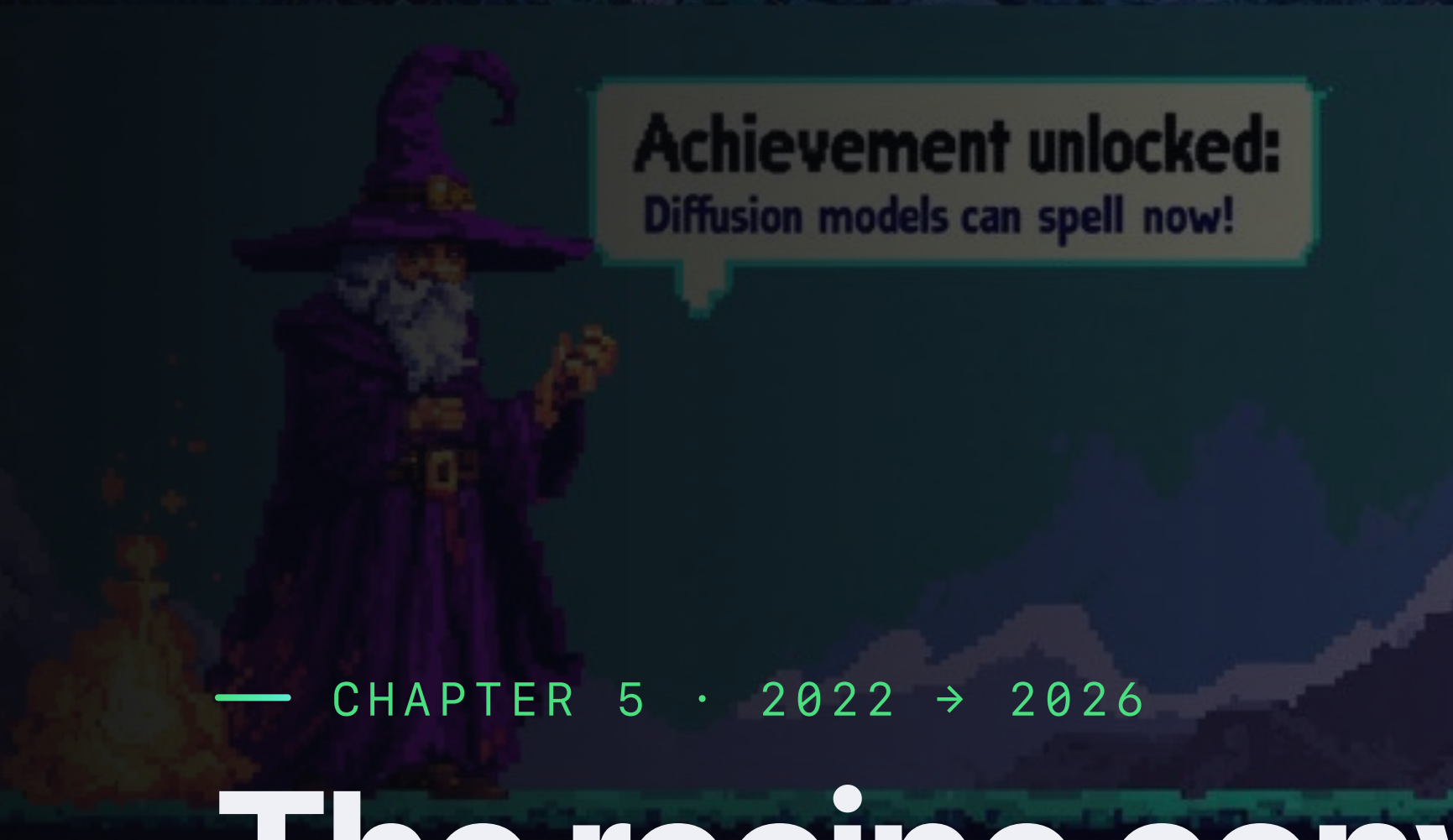


Transformer era: conditions are tokens
OmniGen · Kontext · Qwen-Image-Edit



LoRA never left
FLUX · Qwen · video

➔ Every add-on starts as a no-op on the pretrained model; in Transformers, conditions become tokens.



— CHAPTER 5 · 2022 → 2026

The recipe converges: DiT + flow matching

DiT

flow matching / rectified flow

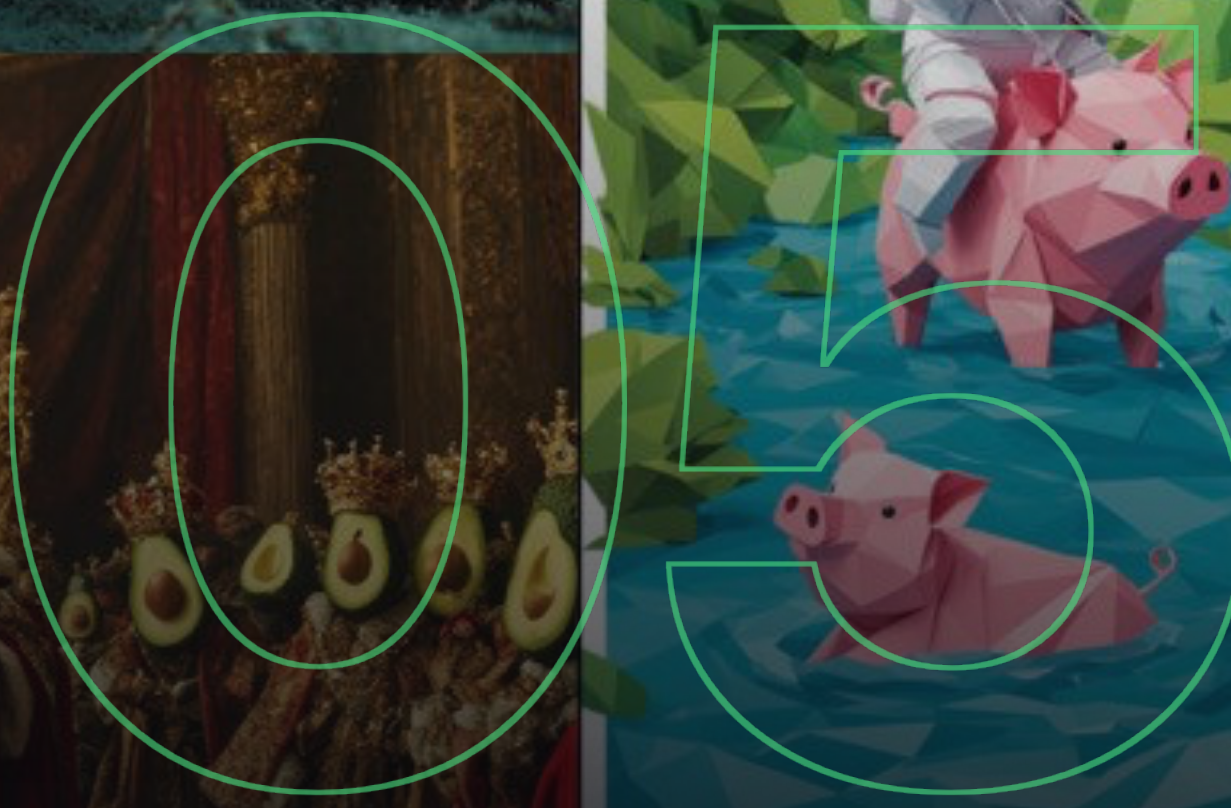
SD3 · MMDiT

FLUX.1

REPA

RAE

JiT



Background: SD3 samples · Esser et al., 2024 · Fig. 1 (crop)

A plain ViT on latent patches **replaces the U-Net**

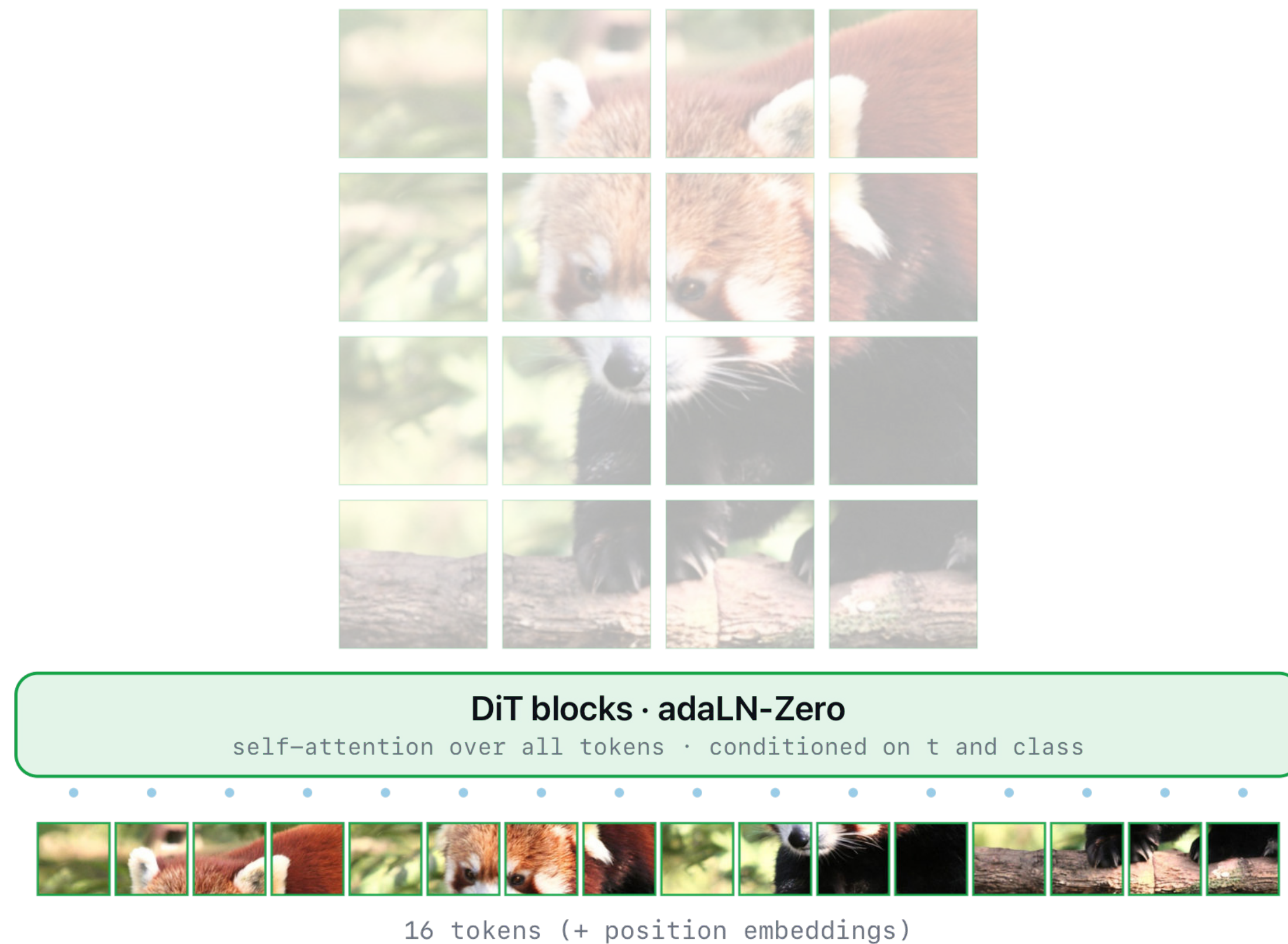
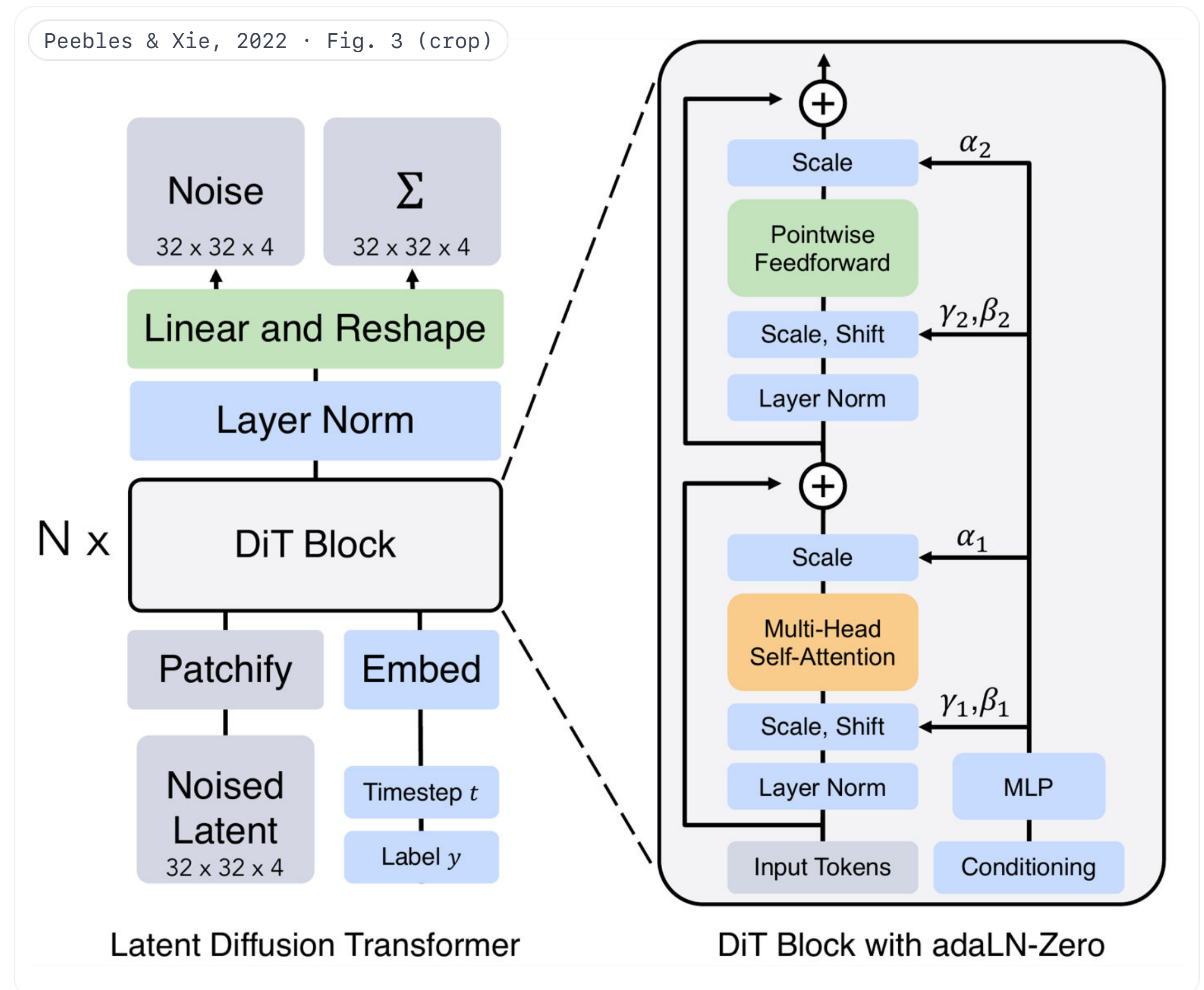


Image: DiT-XL/2 sample · Peebles & Xie, 2022

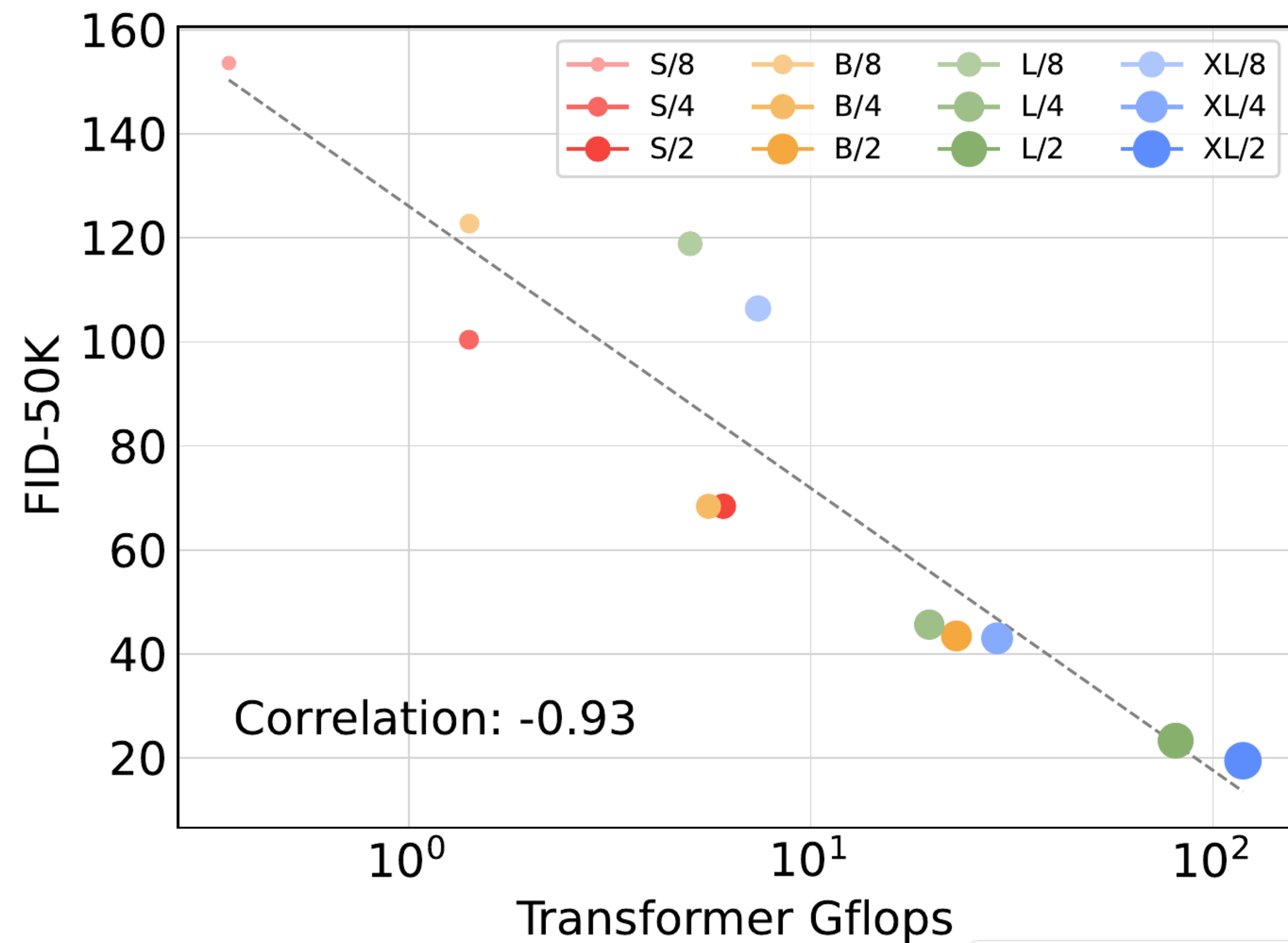
Illustrative · a 256^2 image $\rightarrow$ **$32 \times 32 \times 4$ latent**; $p = 2 \rightarrow 256$ tokens

$\rightarrow$ Diffusion no longer needs a hand-built U-Net: a standard Transformer on latent patches does the job.



adaLN-Zero: $t, \text{class} \rightarrow \text{LayerNorm scale/shift} + \text{zero-init gate}$

Diffusion gets a scaling law: **Gflops predict FID**



Peebles & Xie, 2022 · Fig. 8



Peebles & Xie, 2022 · Fig. 7 (crop)

Same noise · **bigger model** → **smaller patches** ↓

2.27

FID · DiT-XL/2
CFG 1.5

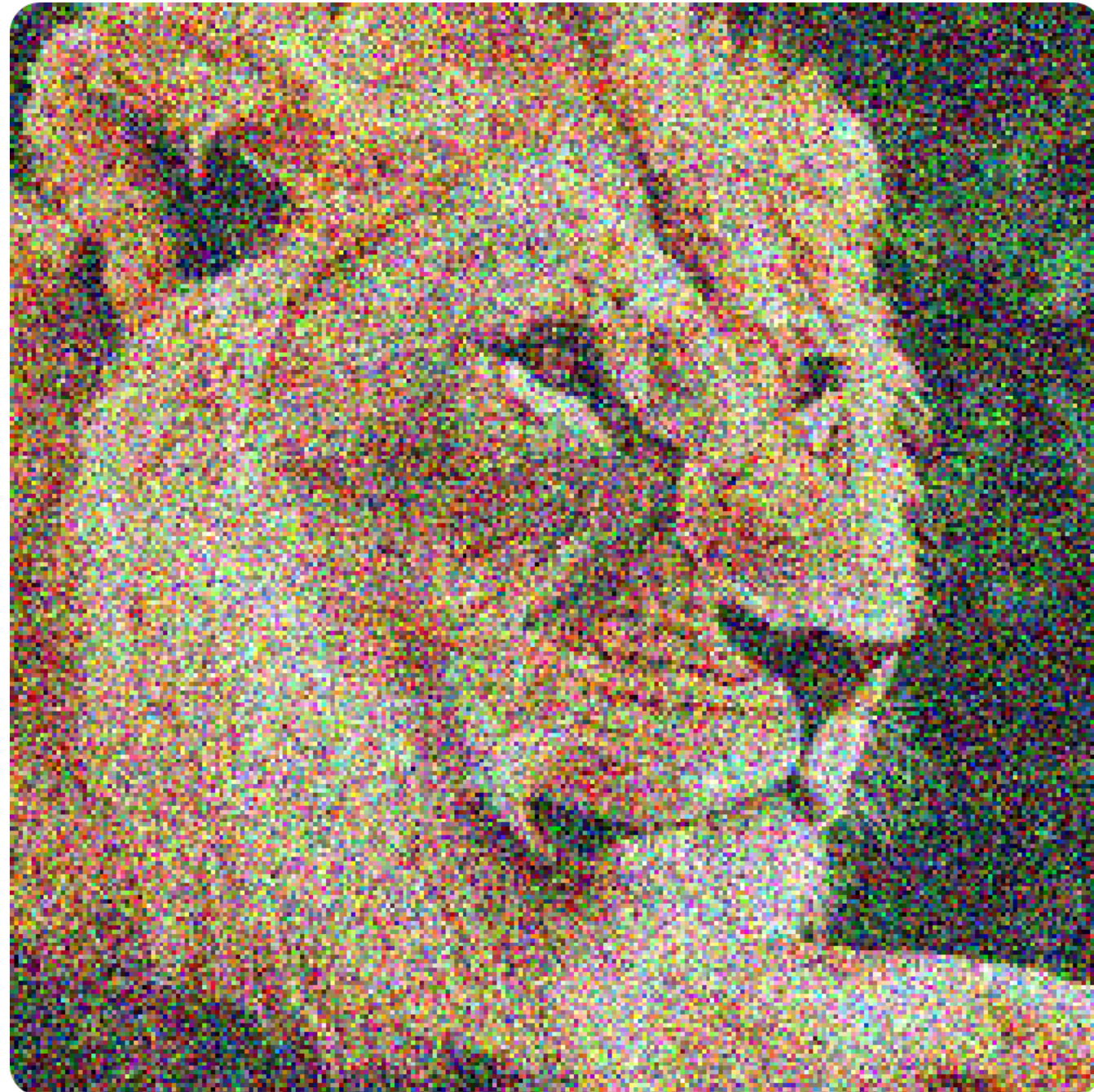
3.60

PREVIOUS BEST
DIFFUSION · LDM

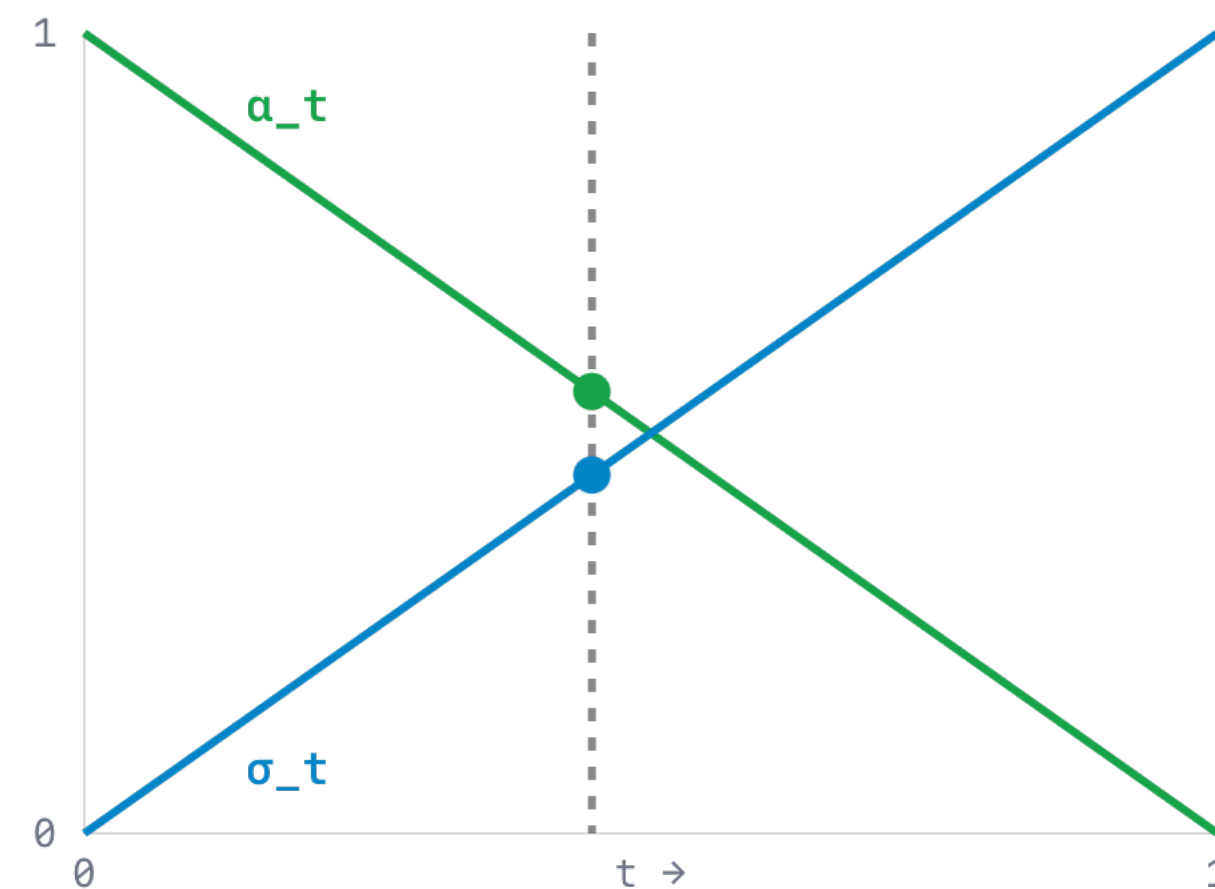
→ Deeper, wider or more tokens all lower FID: diffusion inherits the Transformer scaling playbook.

Flow matching: regress velocity along a straight line

Image: DiT-XL/2 sample · Peebles & Xie, 2022



t 0.45



α_t (signal) = 0.55
 σ_t (noise) = 0.45
 $t = 0.45$ SNR = $\alpha^2/\sigma^2 = 1.5$
linear path: $\alpha = 1 - t$, $\sigma = t$

STRAIGHT PATH · $t = 0$ DATA, $t = 1$ NOISE

$$x_t = (1 - t)x_0 + t\epsilon$$

REGRESS ITS VELOCITY · SIMULATION-FREE

$$\mathcal{L} = \mathbb{E} \left\| v_\theta(x_t, t) - (\epsilon - x_0) \right\|^2$$

It learns the **expected velocity** of the lines through (x_t, t) .

SD3 calls this **rectified flow**.

Rectified Flow

Stochastic Interpolants

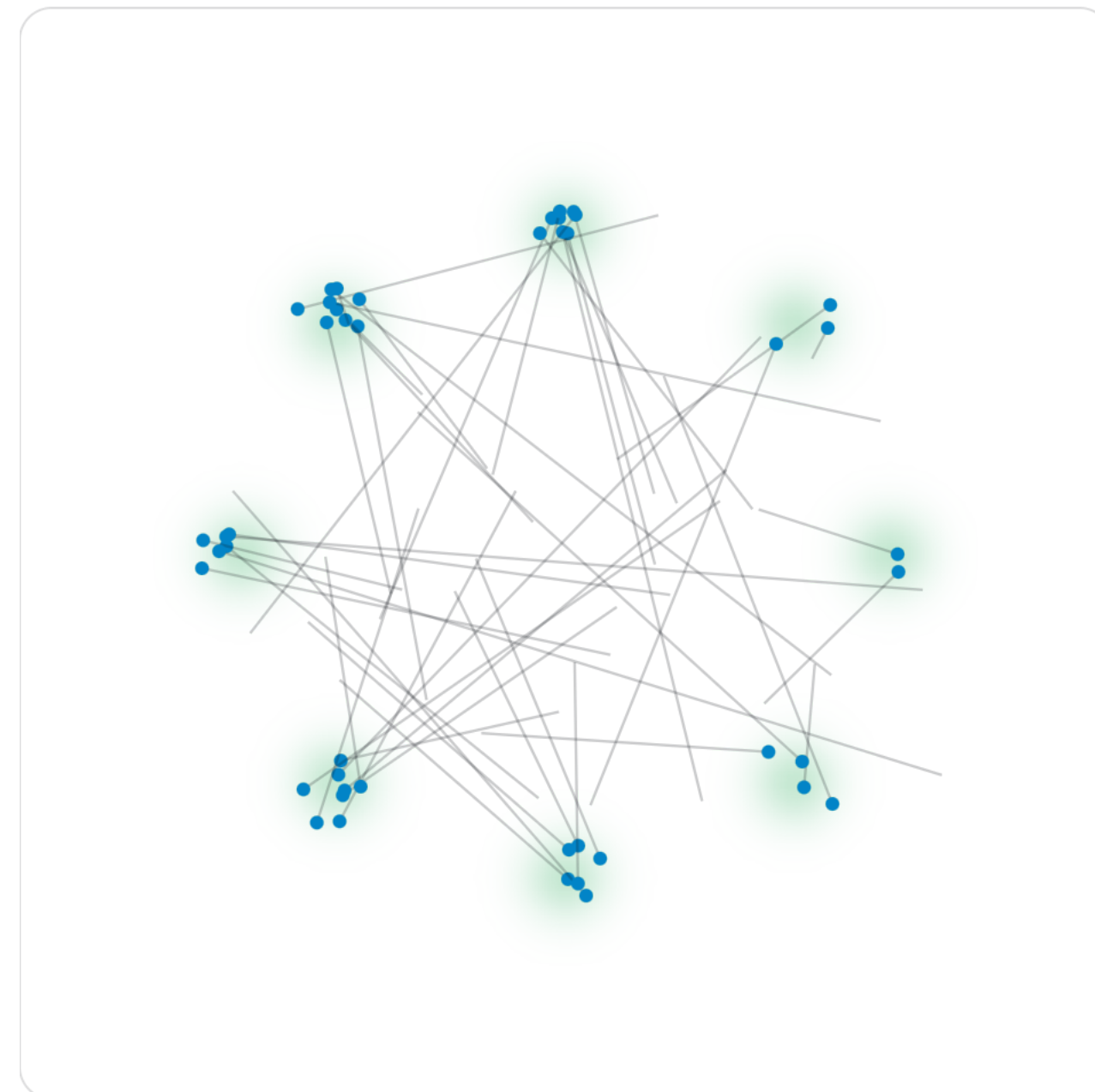
Flow Matching

→ Training needs no ODE simulation: just regression at random points on noise–data lines.

Straight training lines, but the learned flow curves

grey: exact ODE path · colored dots: Euler steps

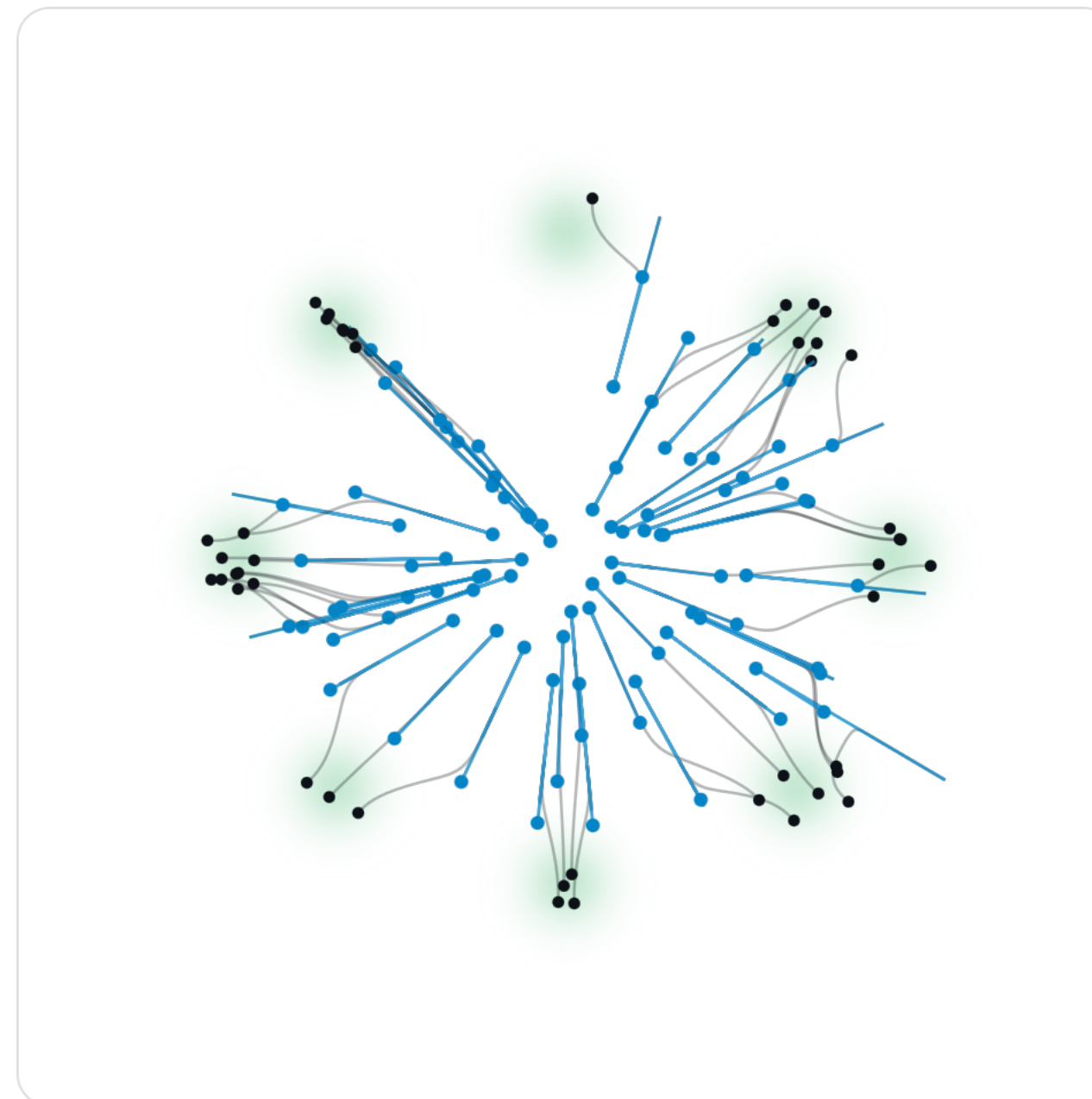
Training pairs: straight, but they cross



$x_t = (1-t) \cdot x_0 + t \cdot \varepsilon$, random pairing

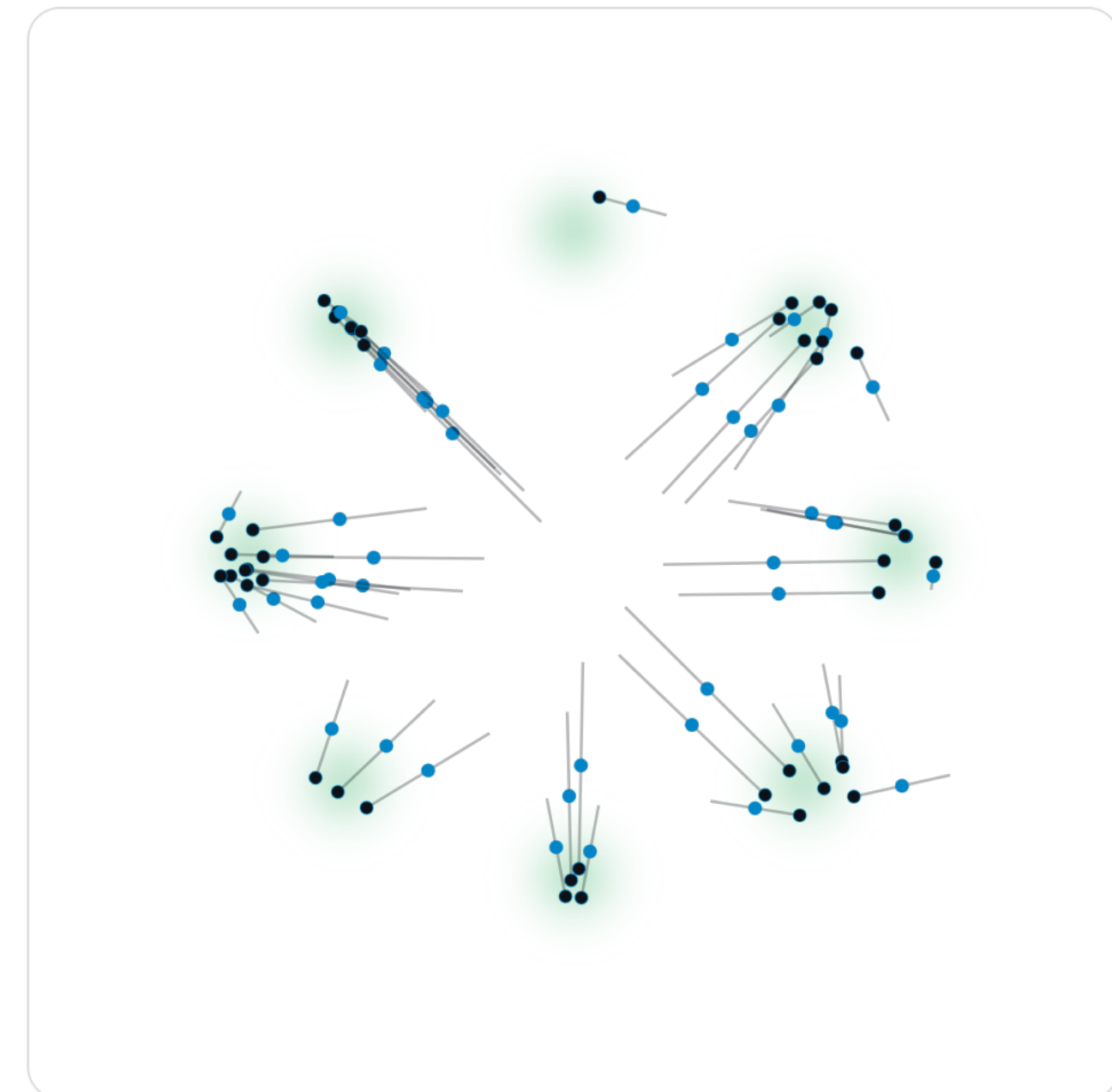
Euler steps

Flow-matching ODE: curved



marginal paths cannot cross

After reflow: straight



idealised 2-rectified flow

2-D toy · reflow: retrain on the model's own noise-image pairs

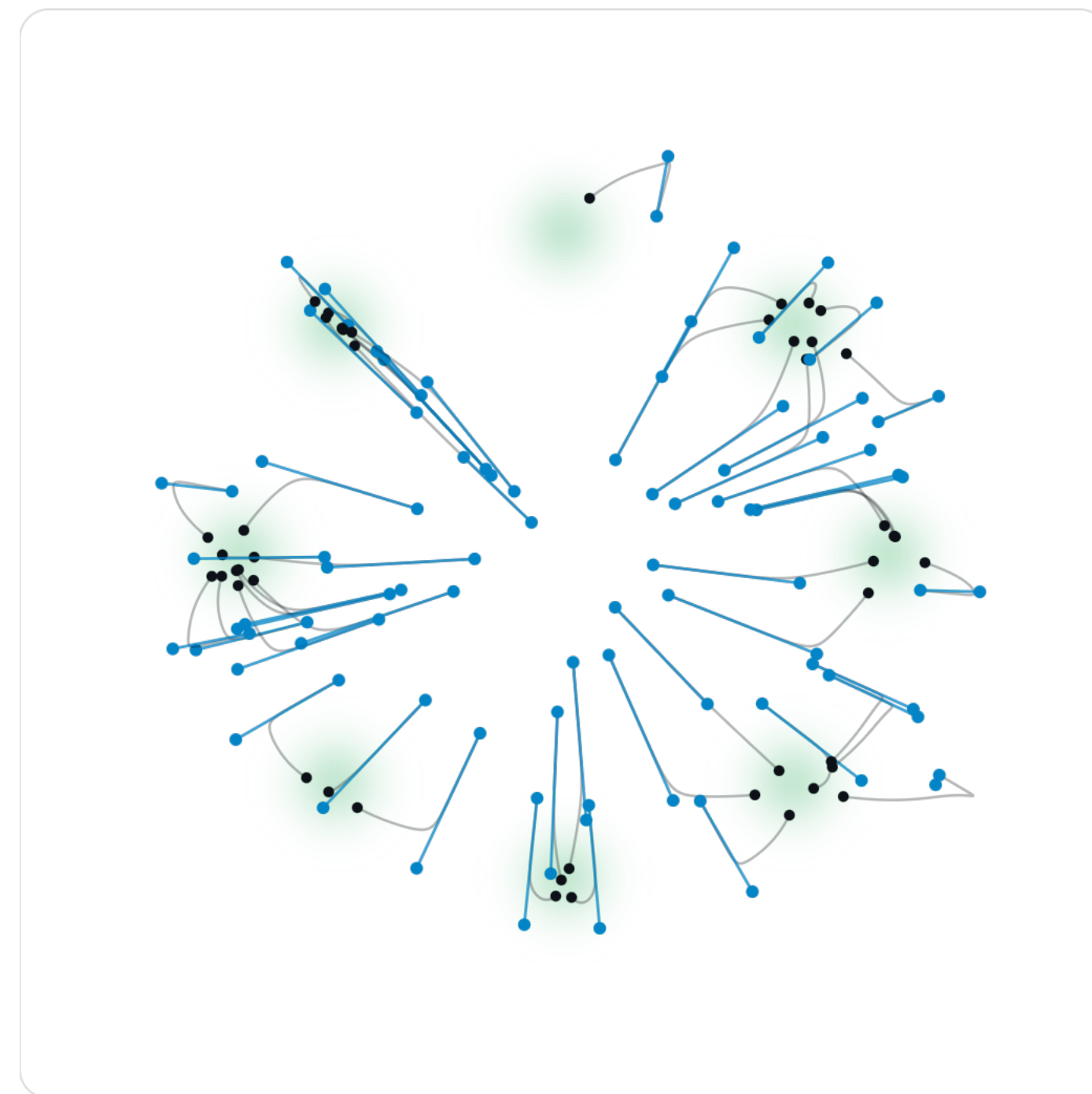
Lines cross, so the averaged velocity bends. Two Euler steps miss on curves, land on straight lines.

→ Straight paths need few steps, but plain flow matching does not learn them; reflow (right) retrains to straighten.

Gaussian FM is diffusion **with a different schedule**

grey: exact ODE path · colored dots: Euler steps

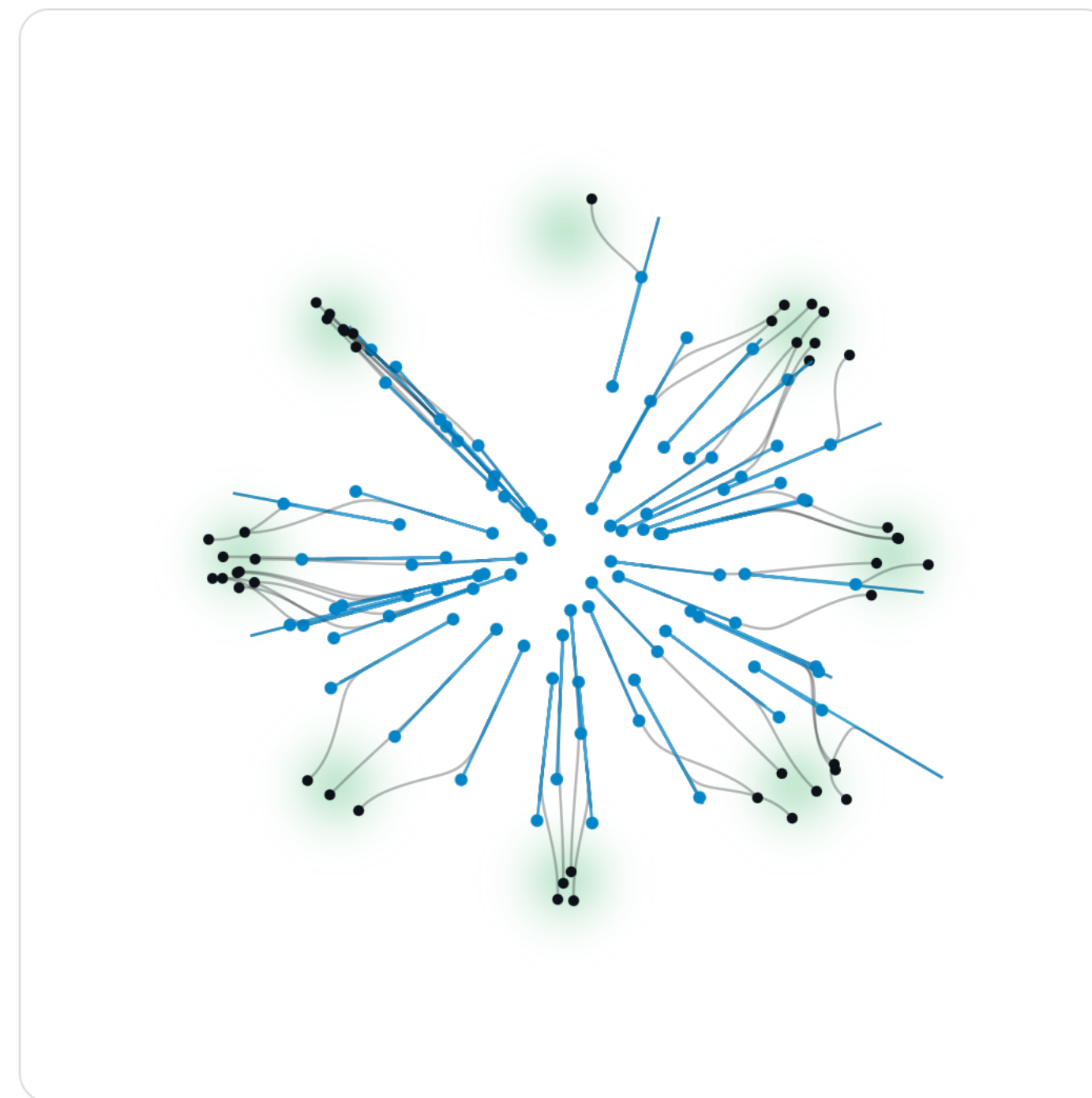
Diffusion PF-ODE (VP, cosine)



exact marginal trajectories

Euler steps 1 2 4 8 32

Flow-matching ODE: curved



marginal paths cannot cross

exact marginal ODE paths · same toy, two schedules

THE STRAIGHT LINE IS A SCHEDULE

$$\alpha_t = 1 - t, \quad \sigma_t = t$$

v IS A REPARAMETERIZATION

$$\hat{x}_0 = x_t - t \hat{v}$$

DDIM = Euler on this schedule.

What matters: **loss weighting** and **output parameterization**.

→ Flow matching did not beat diffusion as a family; it found simpler, better-tuned defaults.

SD3 tests 61 recipes: **rectified flow** + **logit-normal** wins

NO SHIFT (1.0)

Esser et al., 2024 · Fig. 6 (adapted: first 3 prompts)

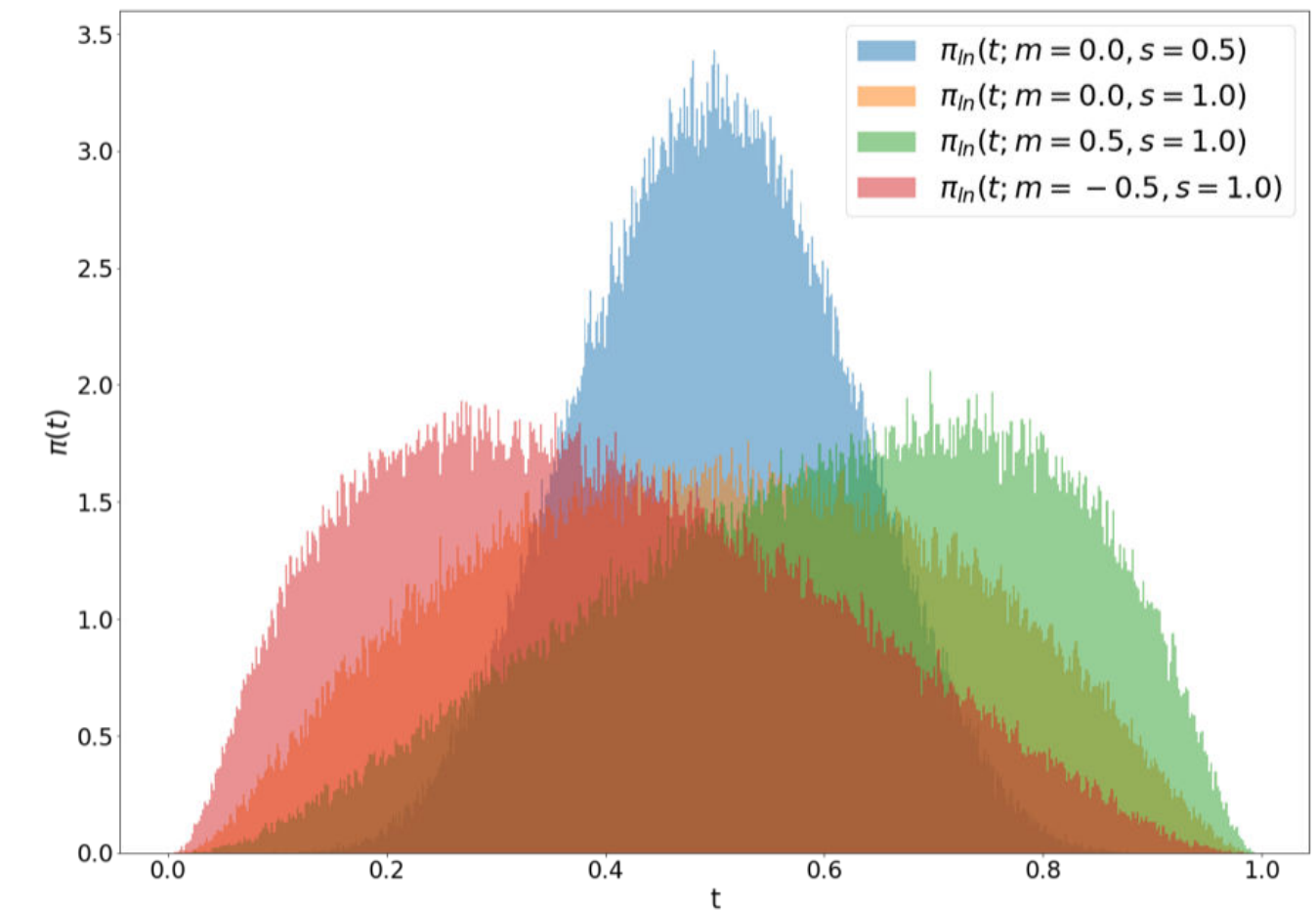


SHIFT 3.0 · SAME PROMPTS



TIMESTEP SAMPLING · TRAIN THE MIDDLE MOST

$$\text{logit}(t) \sim \mathcal{N}(m, s^2)$$



Esser et al., 2024 · Fig. 11 (right)

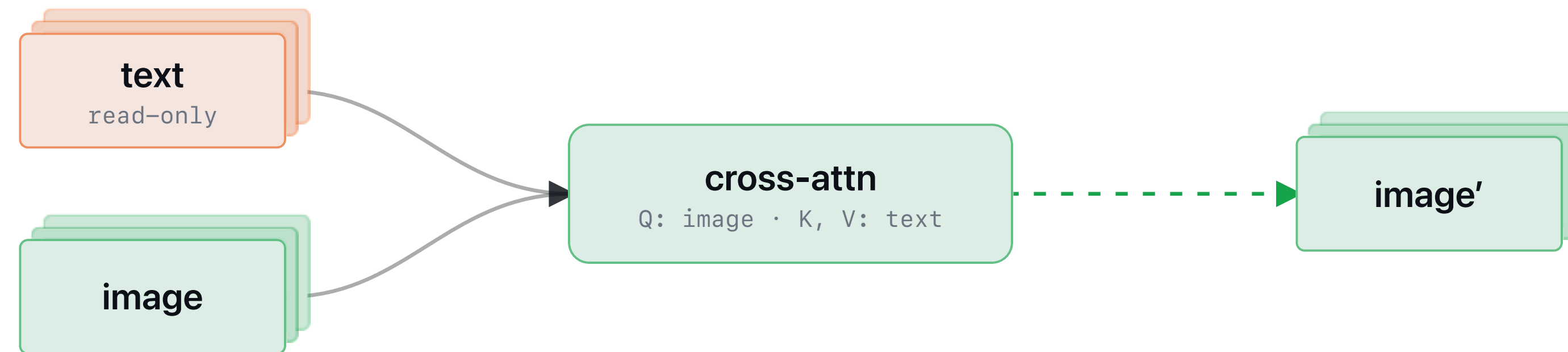
RESOLUTION SHIFT · 3.0 AT 1024²

$$t_m = \frac{\alpha t_n}{1 + (\alpha - 1) t_n}, \quad \alpha = \sqrt{m/n}$$

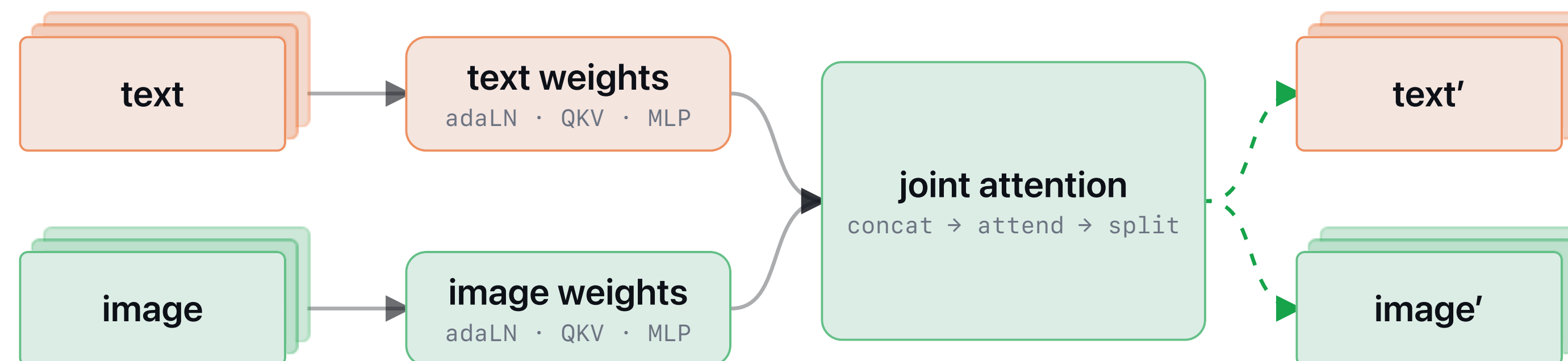
→ Train mid noise levels most and shift toward noise at high resolution; the shift lives on in FLUX and video.

MMDiT: text and image tokens **attend** to each other

Cross-attention · SD 1.x, SDXL



MMDiT · SD3



Esser et al., 2024 · Fig. 1 (crops)

SD3 samples · **legible text in images**

→ Text tokens now update from the image too; per-modality weights beat shared weights and cross-attention.

FLUX.1: SD3's recipe at 12B, **the new open standard**



[pro]

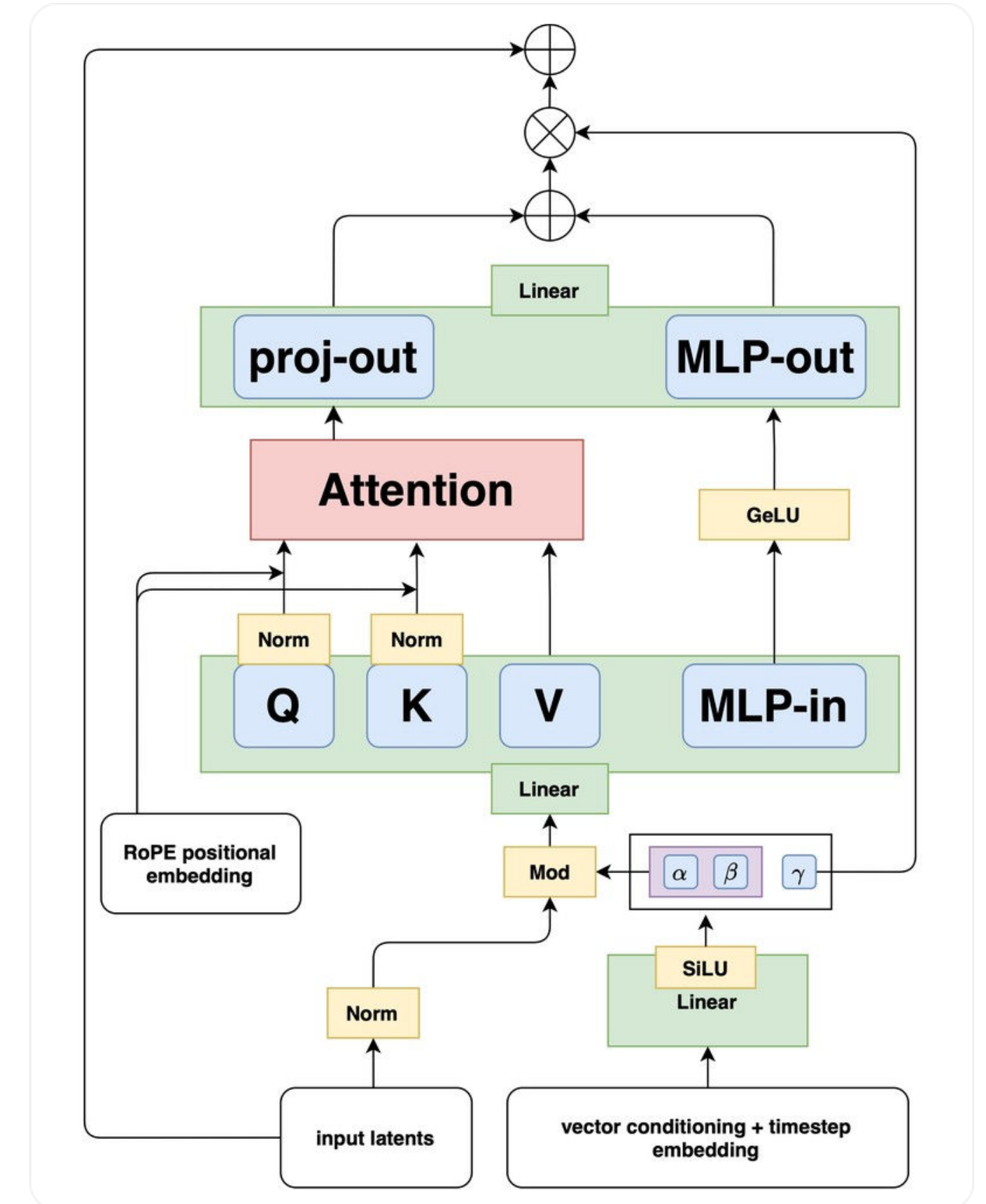
API only

[dev]

open weights · CFG baked in · non-commercial

[schnell]

1–4 steps · Apache 2.0

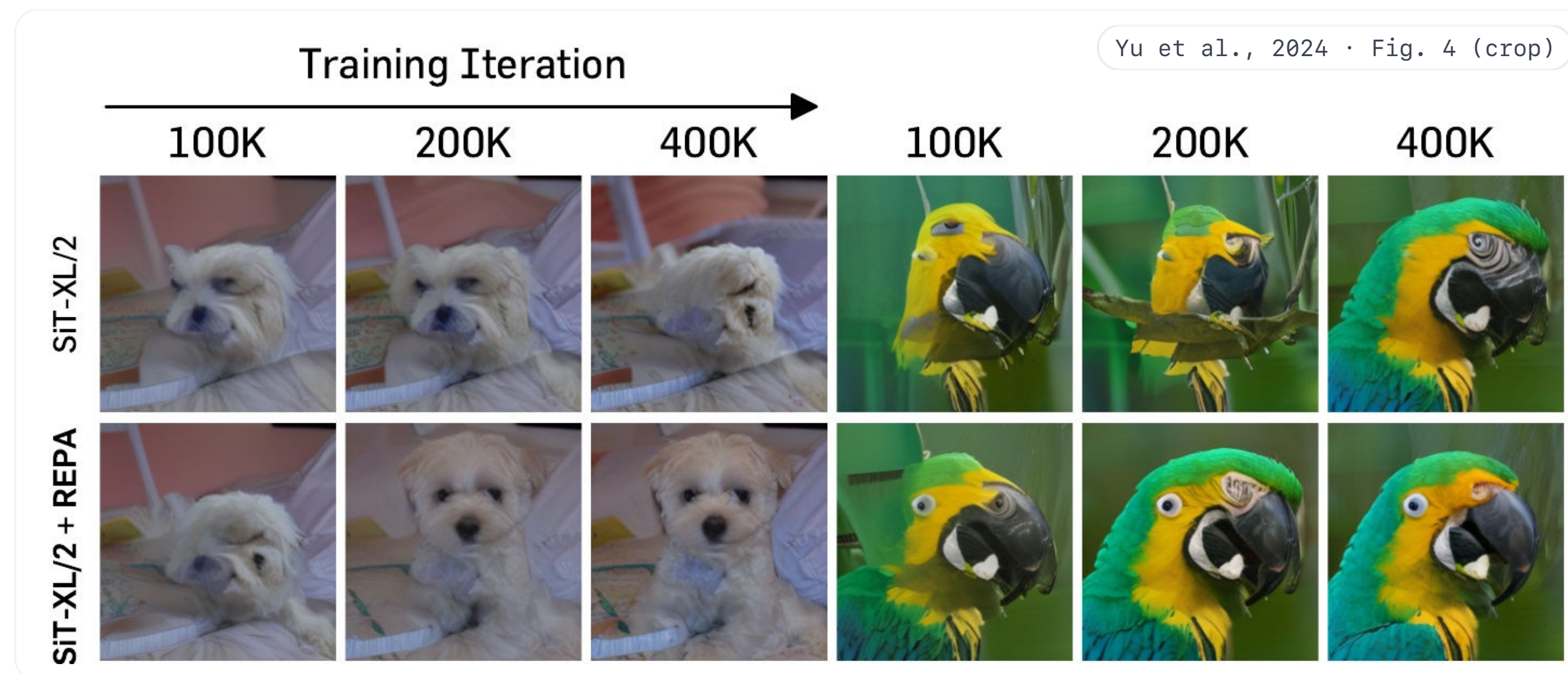


Black Forest Labs, 2025 · Kontext Fig. 3

19 double-stream (MMDiT)
→ **38 single-stream**: shared weights

➔ Technically a refinement of SD3; arguably its impact was the ecosystem, replacing SDXL as the open-weight default.

REPA: align with DINOv2, **train >17.5x faster**

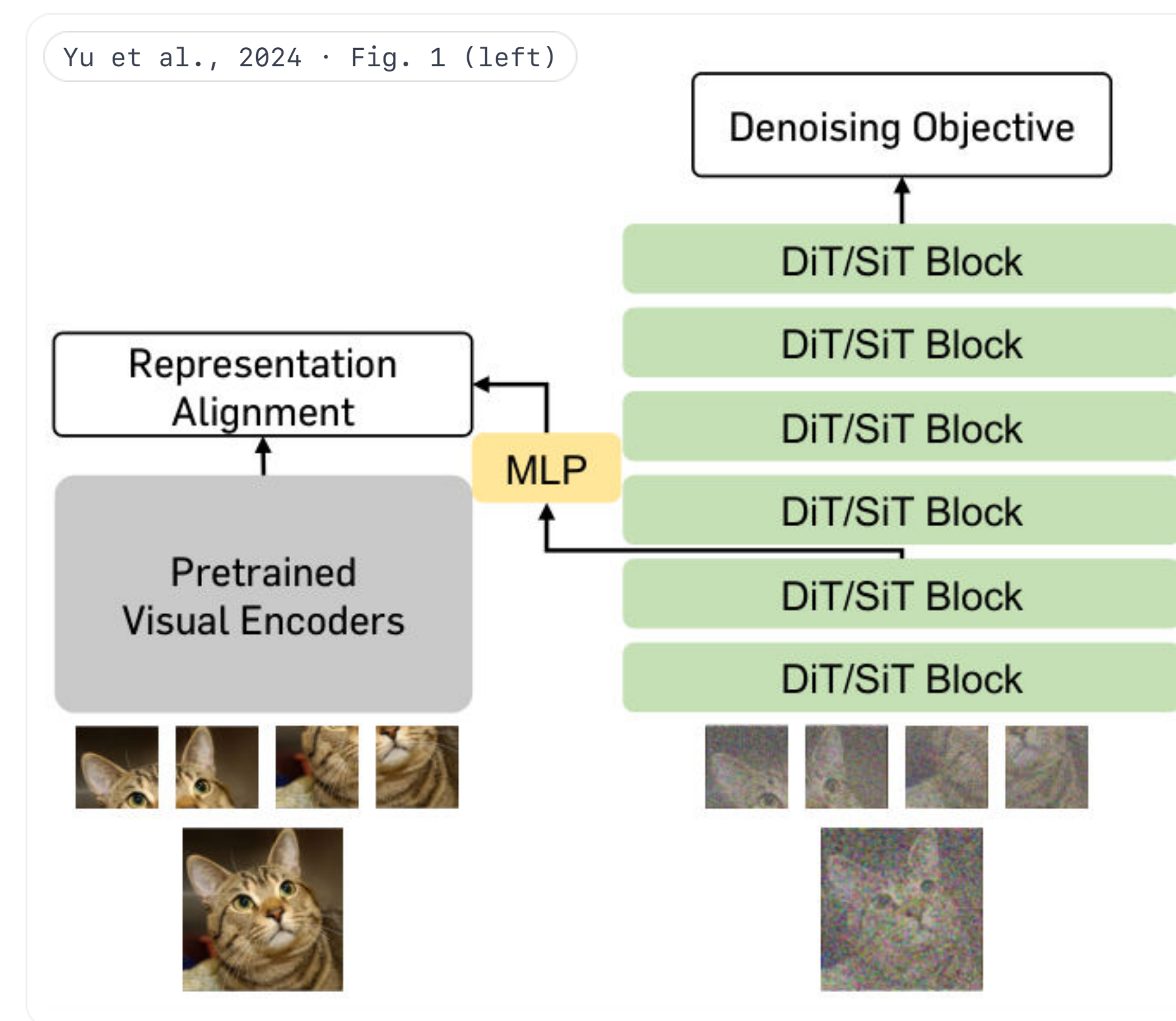


>17.5x

FASTER SiT TRAINING

1.42

FID · WITH GUIDANCE

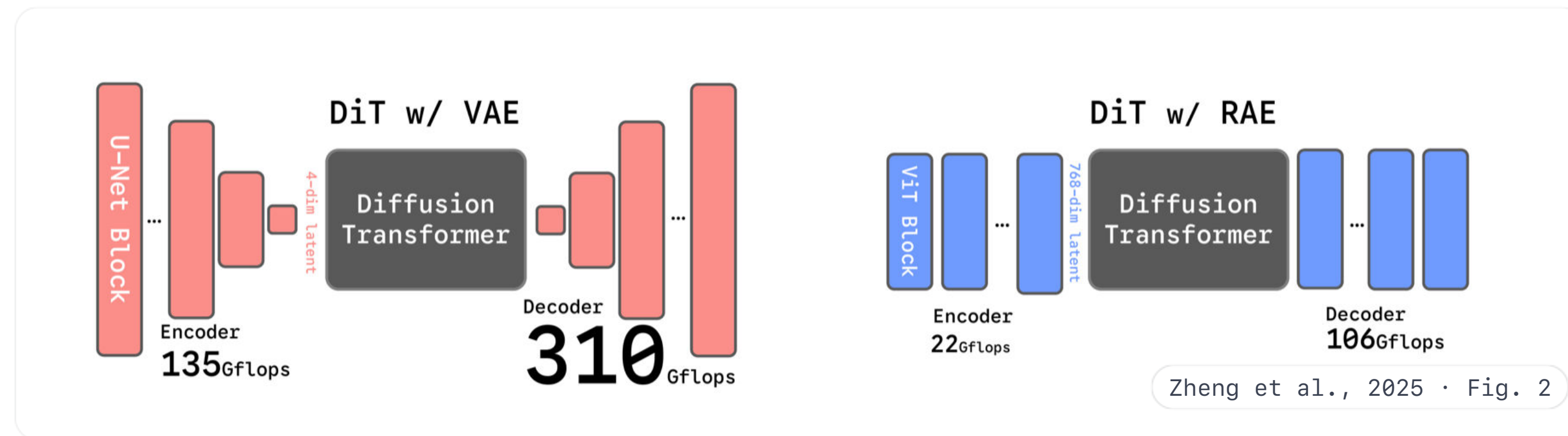


FLOW LOSS – PATCH-WISE COSINE TO DINOv2

$$\mathcal{L} = \mathcal{L}_{\text{flow}} - \lambda \frac{1}{N} \sum_n \cos(y_{\text{DINO}}^{[n]}, h_{\phi}(h_t^{[n]}))$$

→ One reading: much of diffusion training is representation learning, which frozen DINOv2 features can supply.

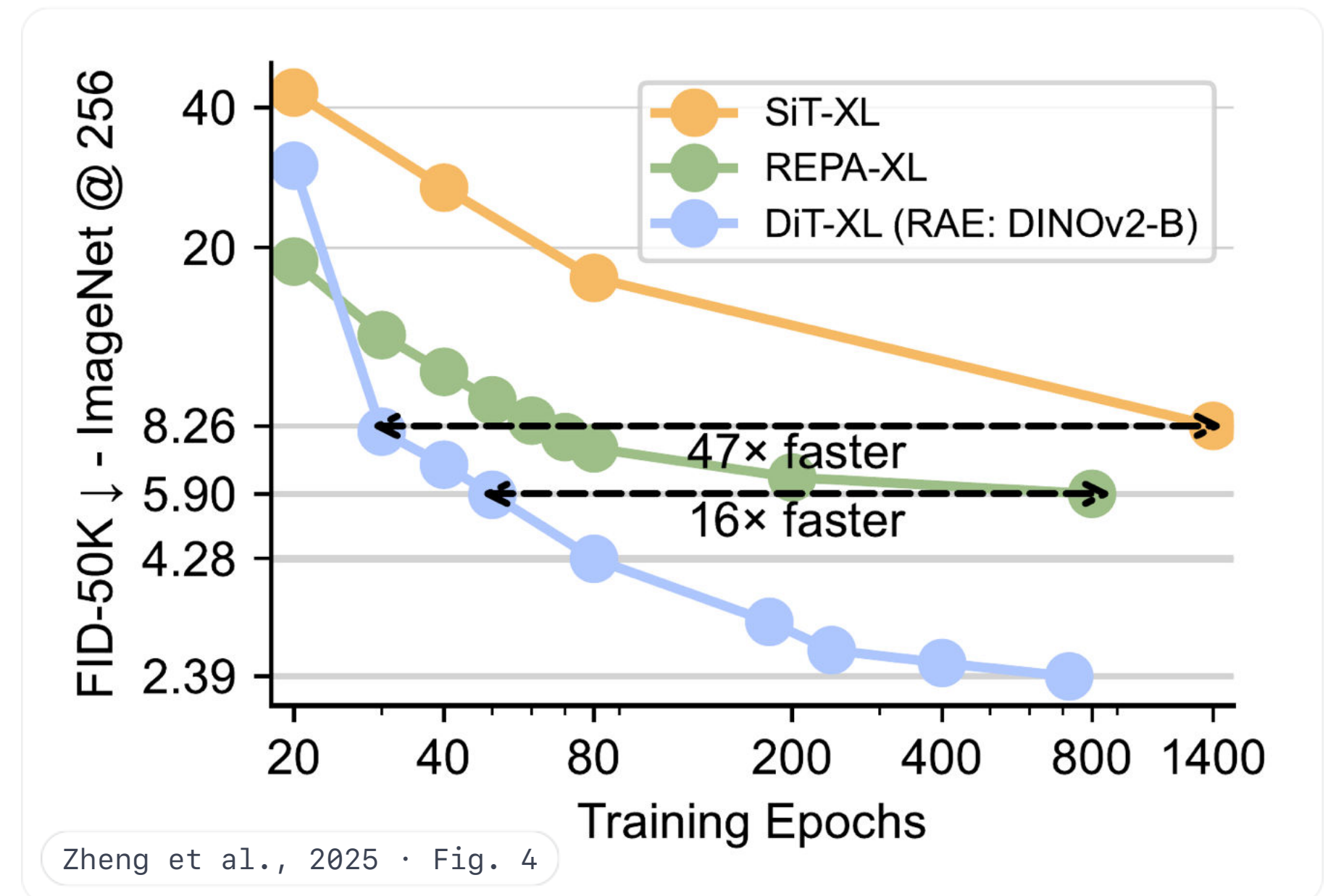
RAE: diffuse in a frozen encoder's semantic latent



DiT width $\geq$ token dim · else it can't fit

Shift by data dim · tokens $\times$ channels

Noise-robust decoder · trained on noised latents

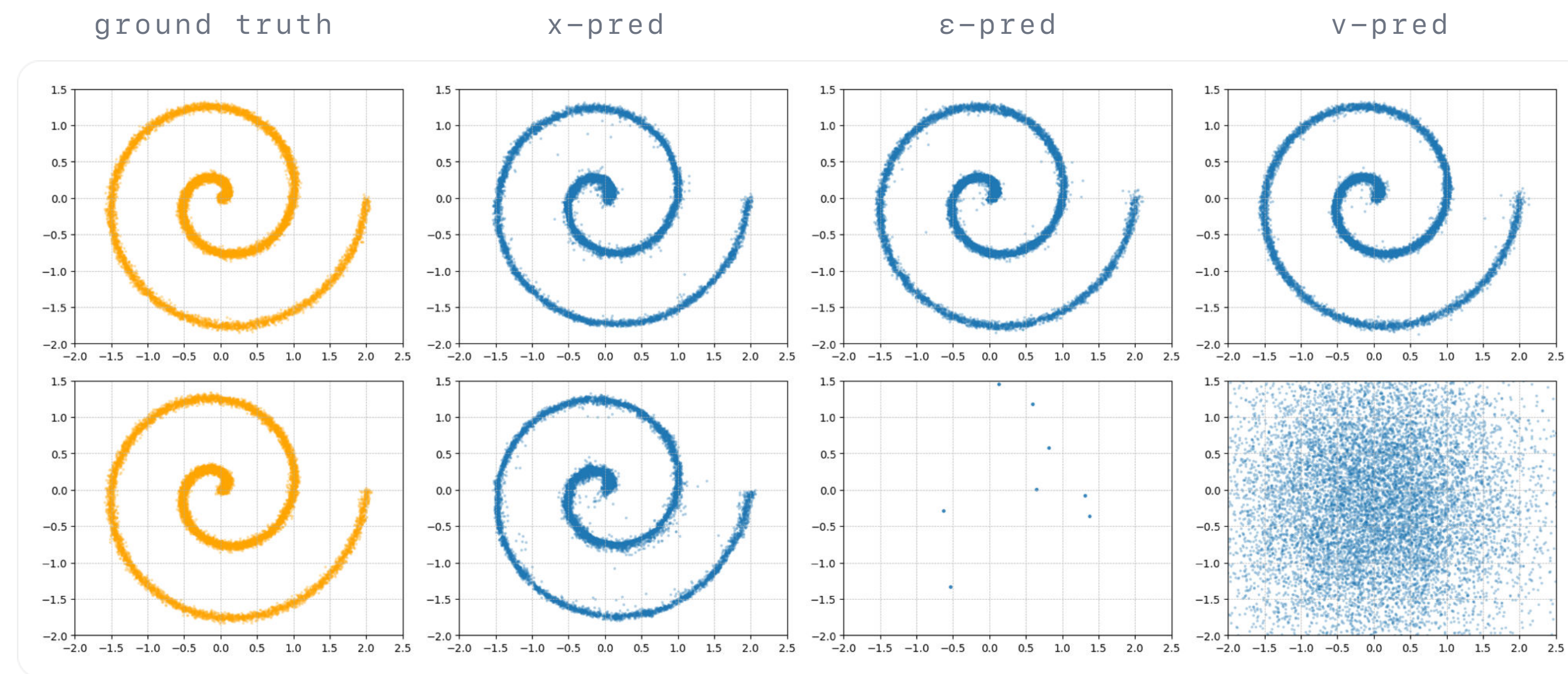


1.13

FID · GUIDED (1.51 UNGUIDED)

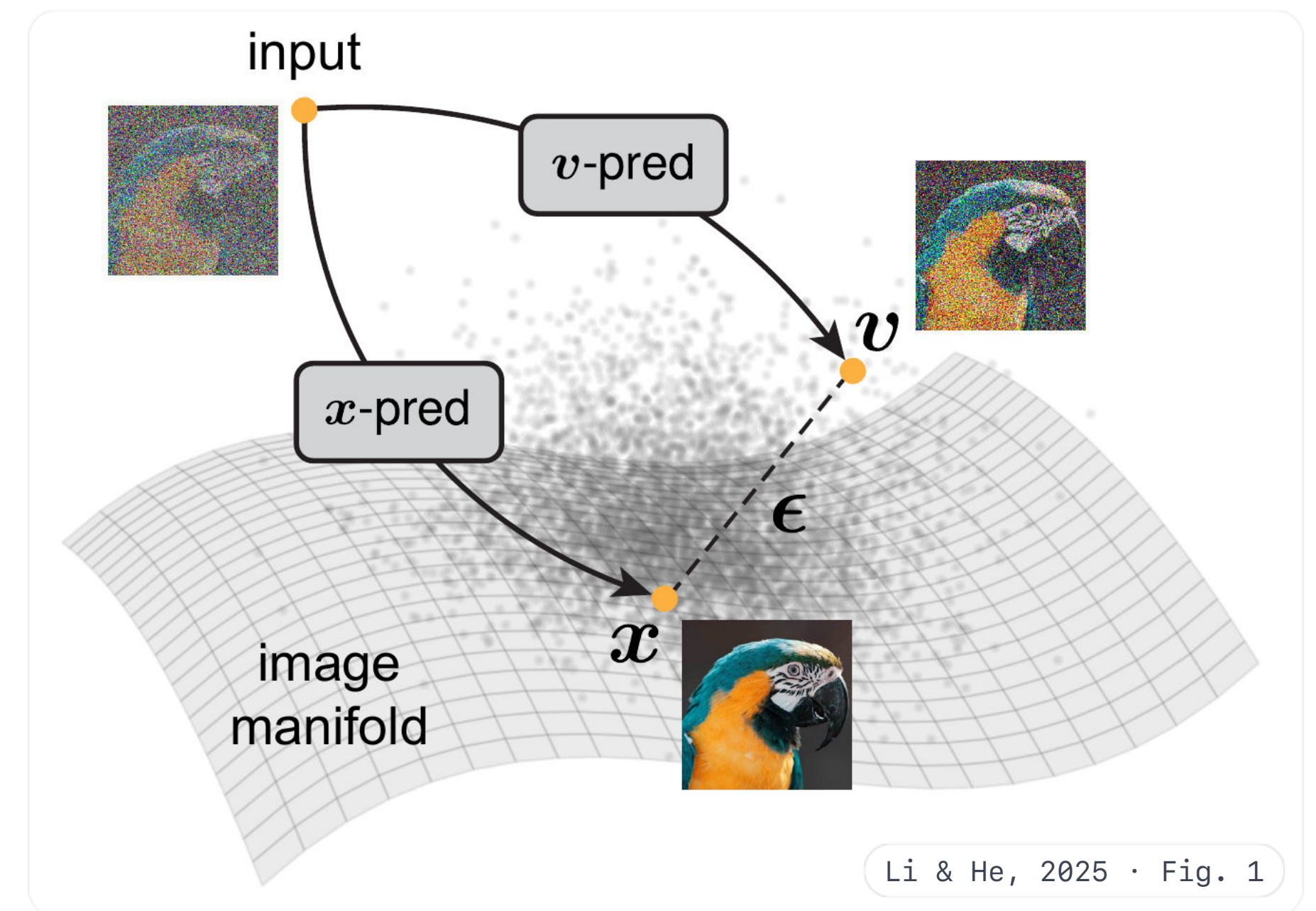
→ The latent need not be a reconstruction VAE: semantic latents converge faster once three fixes are in.

JiT: skip the latent, **predict clean pixels**



Li & He, 2025 · Fig. 2 (adapted: first and last rows)

Top: 2-D data · **bottom:** same data buried in a high-dim space



Li & He, 2025 · Fig. 1

Plain ViT on raw pixels: one $32 \times 32 \times 3$ patch is **3072-dim**.

→ JiT argues: clean images lie on a low-dim manifold, noise does not; so predict x , and a plain ViT works on pixels.

The 2025 default: latent + DiT + rectified flow + LLM

Backbone	U-Net → DiT → MMDiT · single-stream	2022 → 2024
Objective	ε-prediction → velocity (FM / RF) → + logit-normal + shift	2022 → 2024
Text conditioning	cross-attn · CLIP → joint attn · T5 → LLM / VLM encoder	2022 → 2025
Open weights	SDXL → FLUX.1 → Qwen-Image · FLUX.2 · Z-Image	2023 → 2025
Latent	reconstruction VAE → REPA · VA-VAE · REPA-E → RAE (semantic) vs JiT (pixels)	open in 2026

→ A better recipe, not a new model family; still tens of ODE steps per image, which chapter 6 attacks.

— CHAPTER 6 · 2020 → 2026

From 1000 steps to one

A · ODE solvers

B · reflow (straighter paths)

C · progressive distillation

C · Consistency Models

D · ADD · SDXL Turbo

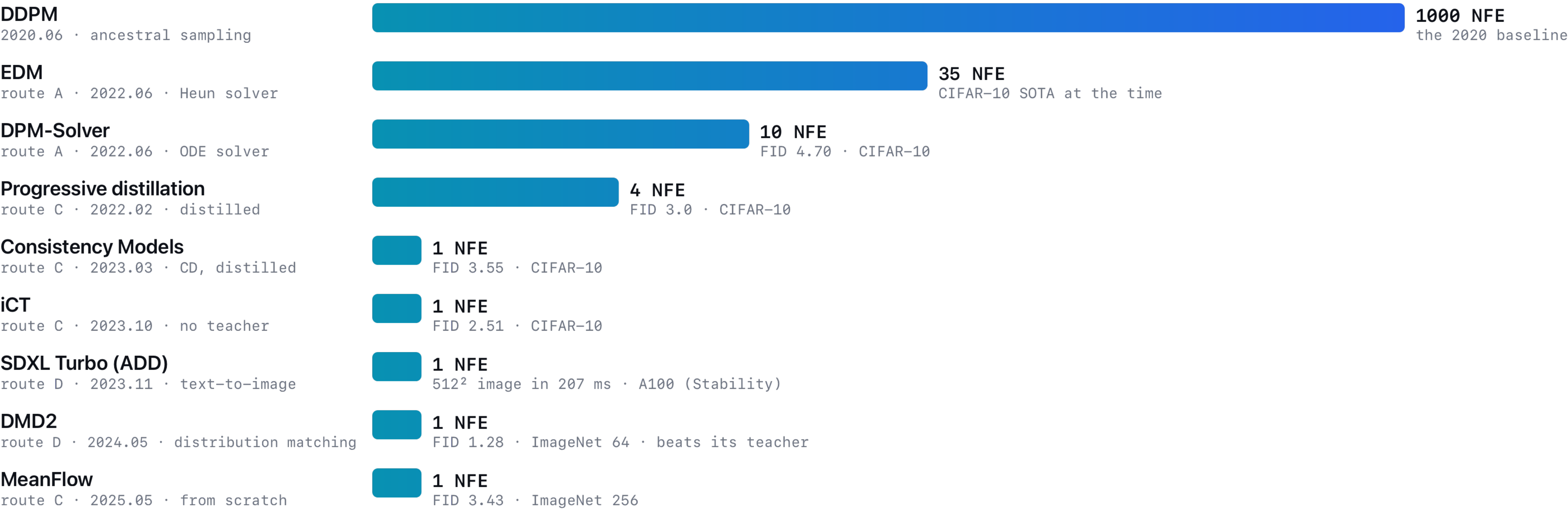
D · DMD

C · MeanFlow

Four routes: A solvers · B straighter paths · C learn to jump · D match distributions

Background: one-step ADD-XL samples · Sauer et al., 2023 · Fig. 1

Five years, 1000× fewer network calls



log scale

Datasets differ (CIFAR-10 · ImageNet 64 / 256): **compare steps, not FIDs** · numbers from each paper or official release

→ Solvers stall near 10 steps; getting to 1 step meant training a new network.

Smarter integrators: 1000 steps to **10–20**



DDIM 10 · 20 · 100 NFE | **DPM-Solver** 10 NFE

Lu et al., 2022 · Fig. 1 (adapted)

DDIM · PREDICT CLEAN,
RE-NOISE TO NEXT LEVEL

$$x_s = \alpha_s \hat{x}_0(x_t) + \sigma_s \hat{\epsilon}(x_t)$$

DPM-Solver++ · stable at high CFG

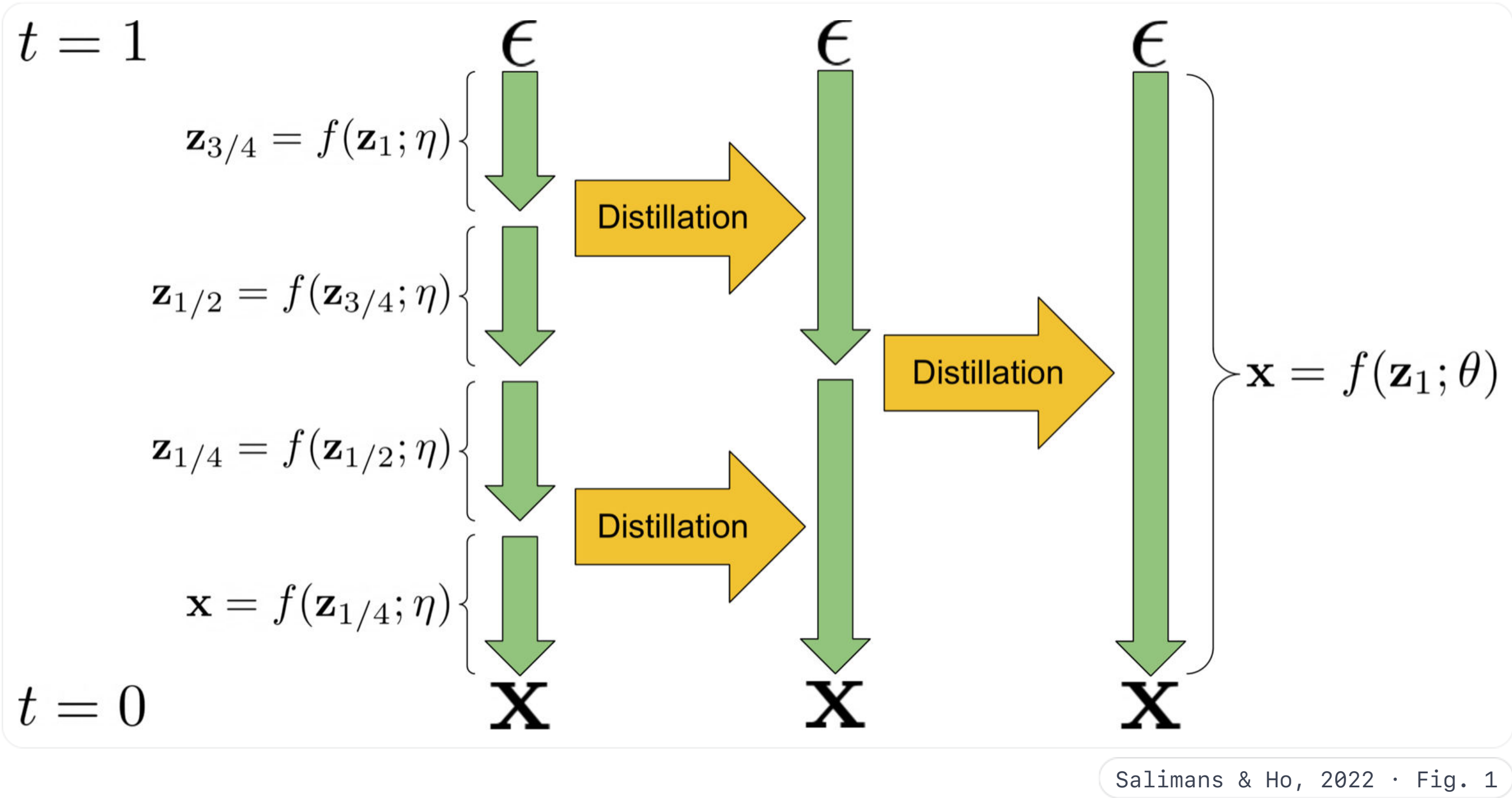
EDM · Heun, 35 NFE

UniPC · Wan2.1 default

below ~10 steps: curved paths → blur

→ Same network, better integrator: no retraining, but curved paths cap it near 10 steps.

Distill two teacher steps into **one**, repeat



8192 → 4

STEPS, HALVED EACH ROUND
4-STEP FID 3.0 · CIFAR-10

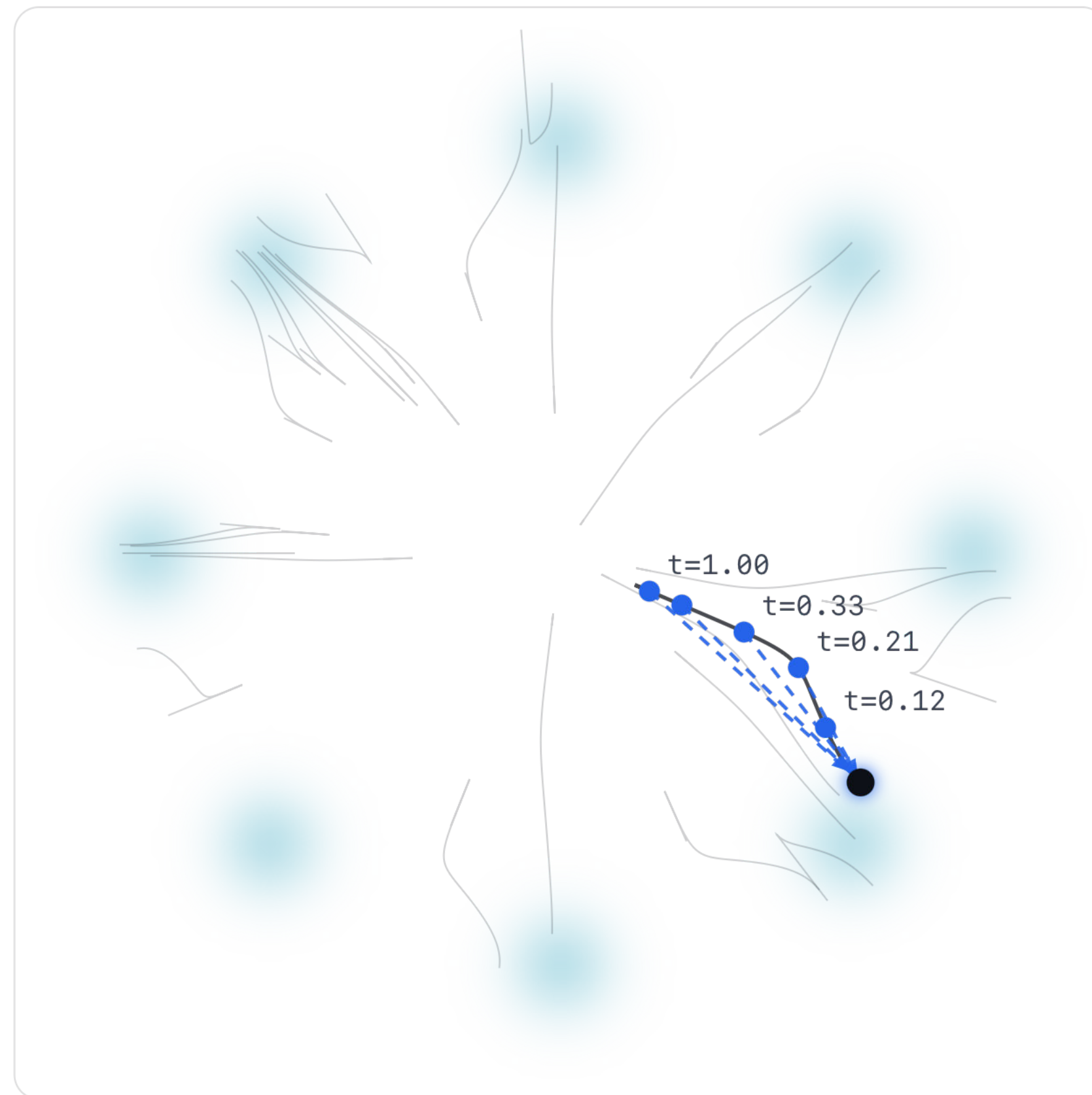
RECALL PRIMER · V-PREDICTION, BORN HERE

$$v = \alpha_t \epsilon - \sigma_t x$$

v adopted: SD 2.x 768, Imagen Video
Guided distillation (2022.10) bakes CFG into one pass
FLUX.1 [dev]: “guidance distillation” [details undisclosed](#)

→ Each round halves the step count, but every round retrains the student.

Learn the trajectory's **endpoint**, not its tangent



$f_{\theta}(x_t, t) = x_0$
for every t on the
same ODE trajectory

toy 2-D PF-ODE · after Song et al., 2023 · Fig. 1



top EDM teacher, many steps · **bottom** CD, 1 step · same noise

Song et al., 2023 · Fig. 16 (adapted)

SAME OUTPUT ALONG A PATH · IDENTITY AT THE START

$$f_{\theta}(x_t, t) = f_{\theta}(x_{t'}, t'), \quad f_{\theta}(x, t_{\min}) = x$$

CD (distillation): teacher takes one ODE step

CT (training): no teacher

→ One network call maps any noisy point on a trajectory directly to its clean endpoint.

Teacher-free CT beat distillation; LCM went plug-in



top PaperCut style LoRA · **bottom** + LCM-LoRA · SDXL, 2 → 32 steps

Luo et al., 2023 (LCM-LoRA) · Fig. 3 (adapted)

2.51

1-STEP FID · CIFAR-10
iCT, NO TEACHER

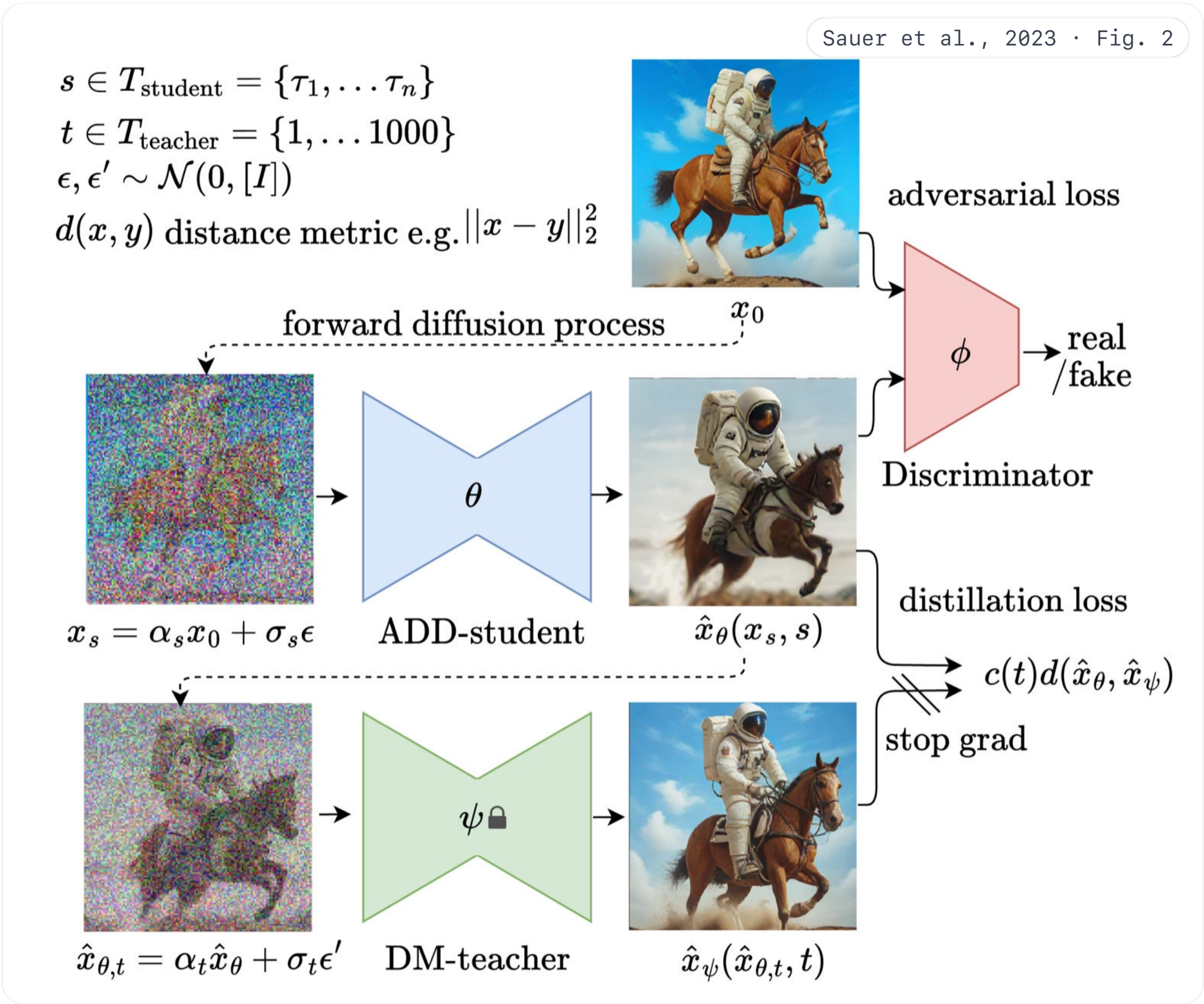
LCM / LCM-LoRA

SD in 2–4 steps after 32 A100 hours

DALL·E 3 decoder: CD-distilled to 2 steps

→ Training without a teacher soon beat distillation, and a plug-in LoRA let Stable Diffusion sample in 2–4 steps.

A teacher plus a discriminator train a **one-step** student



50 → 1
STEPS · SDXL TURBO
PER STABILITY

207 ms
512² IMAGE · A100
PER STABILITY



LCM-XL (consistency-distilled SDXL) 1 · 2 · 4 steps → **ADD-XL 1 step**

Sauer et al., 2023 · Fig. 3 (adapted)

Blind test: 1 step beats 4-step LCM-XL; 4 steps beat 50-step SDXL

→ Sharp one-step images came from matching the output distribution, not the teacher's path.

One step or many? **Hard to tell**

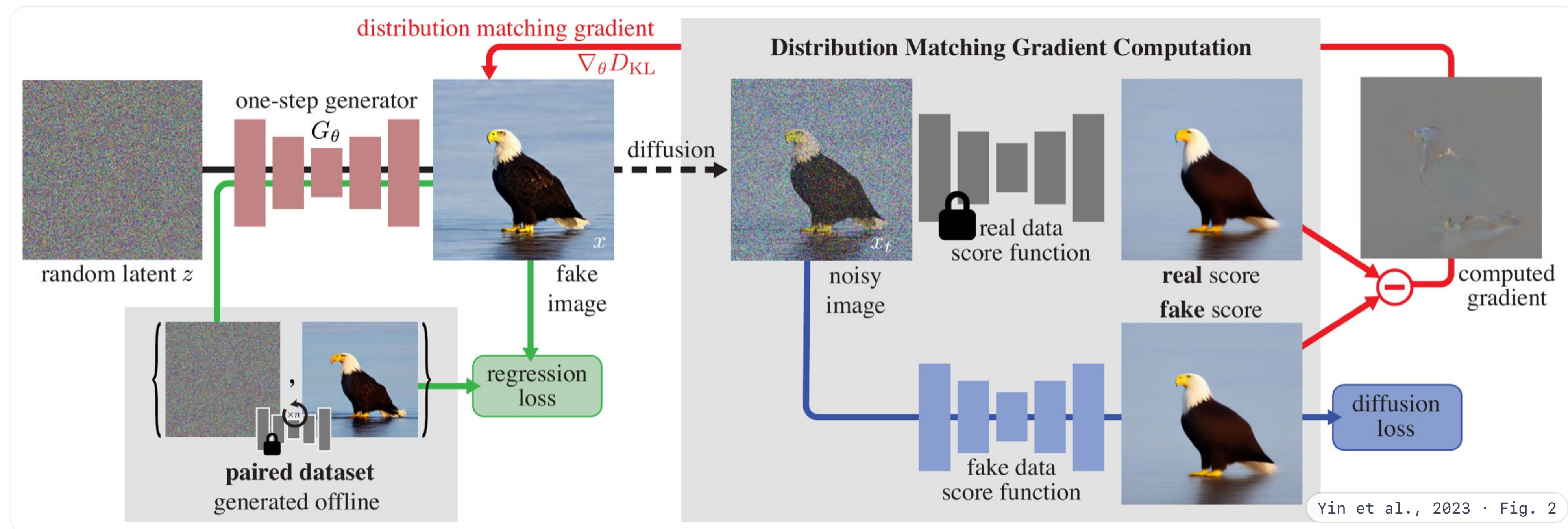


Red boxes: one-step **DMD**. The other image in each column: multi-step Stable Diffusion

Yin et al., 2023 · Fig. 1

→ By late 2023, one network call could pass for a multi-step Stable Diffusion image.

Real score minus fake score **is the gradient**



REVERSE KL · SCORES ON RE-NOISED GENERATOR OUTPUT

$$\nabla_\theta D_{KL} \approx \mathbb{E}_{z,t} \left[- \left(s_{\text{real}}(x_t, t) - s_{\text{fake}}(x_t, t) \right) \frac{\partial G_\theta(z)}{\partial \theta} \right]$$

2.62

1-STEP FID
IMAGENET 64

11.49

FID · ZERO-SHOT
COCO-30K

→ The teacher says where real images are dense; a second network tracks where the student over-clusters.

Turbo products run on **distribution matching**



Samples: Yin et al., 2024 · DMD2 (4-step SDXL)

1.28

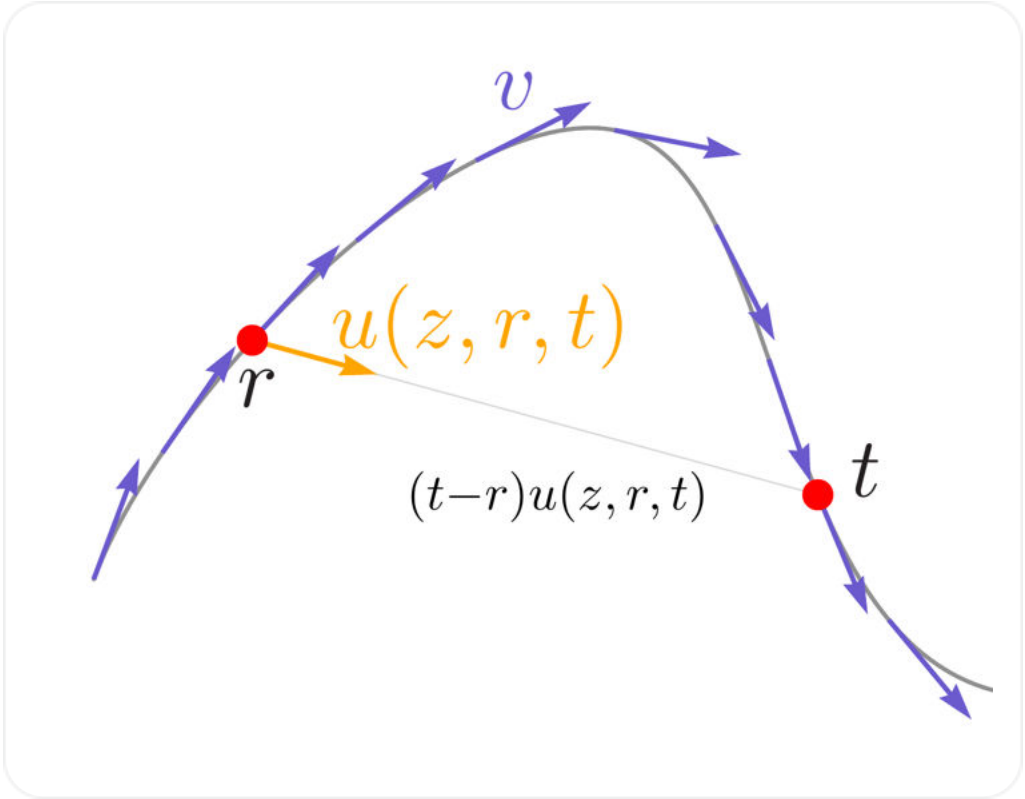
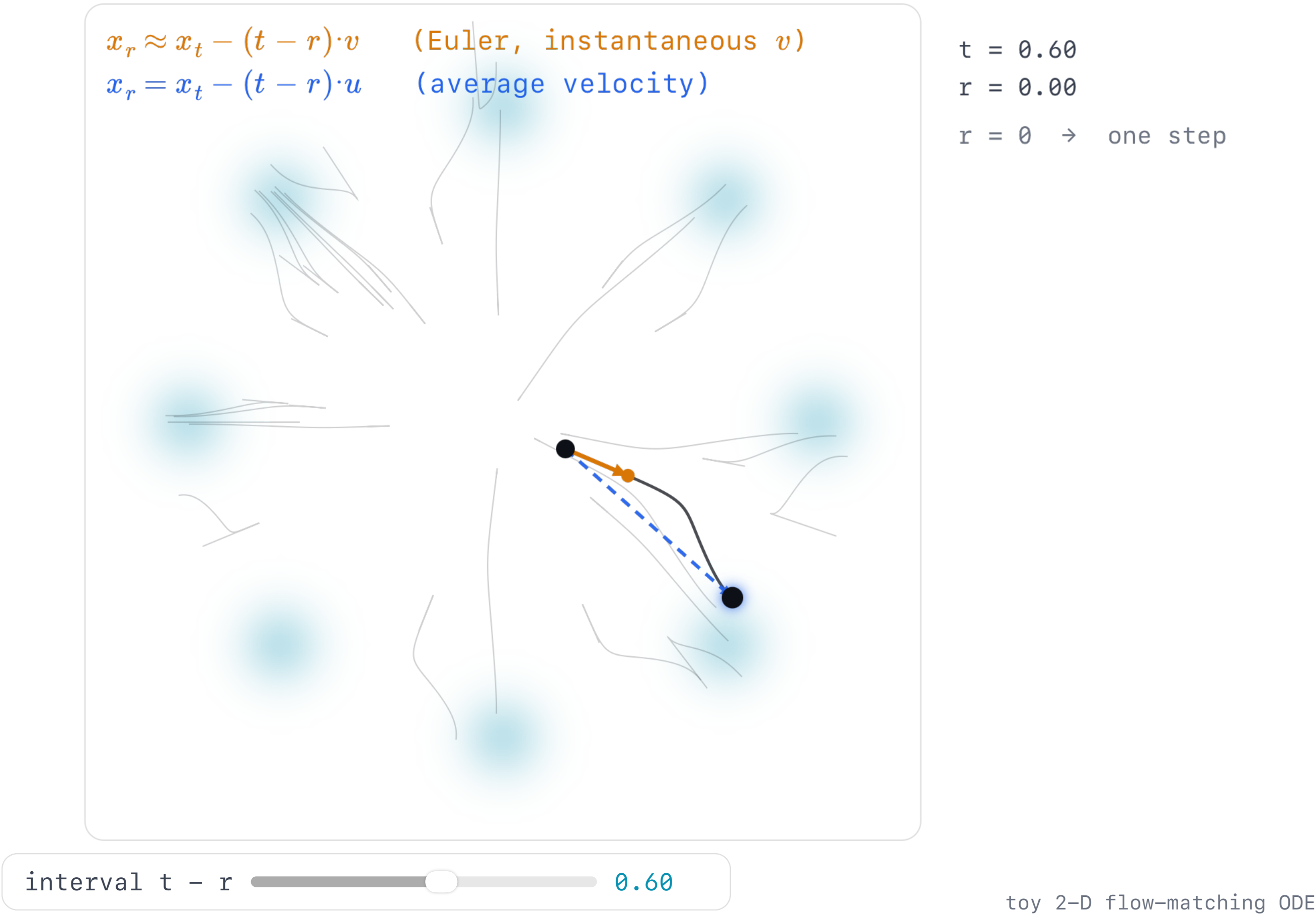
1-STEP FID · IMAGENET 64
DMD2 BEATS ITS TEACHER

- SD3-Turbo · LADD · 4 steps
- FLUX.1 [schnell] · 1-4 steps
- SDXL-Lightning
- Z-Image-Turbo · 8 NFE

Real-time video: CausVid, Self-Forcing → ch8

→ Matching the output distribution, not the teacher's path, became the main recipe for 1-4-step products.

Learn the **average velocity**, jump in one step



Geng et al., 2025 · Fig. 3 (left)

AVERAGE VELOCITY OVER AN INTERVAL

$$u(z_t, r, t) = \frac{1}{t - r} \int_r^t v(z_\tau, \tau) d\tau$$

MEANFLOW IDENTITY · ONE JVP

$$u = v - (t - r) \frac{d}{dt} u$$

→ Learn the average velocity over a whole interval, and one big jump replaces many small steps.

One step **from scratch**, no teacher



1-NFE samples · MeanFlow-XL/2

Samples: Geng et al., 2025 (1-NFE, curated)

3.43

FID · 1 STEP
XL/2 · 240 EP

2.20

FID · 2 STEPS
XL/2+ · 1000 EP

Follow-ups: **iMF** 1.72 at 1 step · **pMF** 2.22 in pixels

validated mainly on ImageNet · at scale: open

➔ From-scratch one-step models sharply narrowed the gap on ImageNet; replacing distillation is open.

Four routes, two concept shifts

A · Solvers	DDIM → DPM-Solver++ · EDM → UniPC	2020 → 2023
B · Straighter paths	reflow (2-rectified flow) → InstaFlow	2022 → 2023
C · Learn to jump	PD → CM · LCM → iCT → MeanFlow	2022 → 2025
D · Match distributions	ADD · DMD → DMD2 · LADD → Turbo products	2023 → 2025
Divergence	CM: diverse, soft + DMD: sharp, collapse-prone → rCM: both, 14B	2025

➔ Learn the jump, not the tangent; match the distribution, not the path.

— CHAPTER 7 · 2022 → 2024

From image models to spacetime patches

VDM · factorized 3D U-Net

Video LDM · temporal layers

Sora · spacetime-patch DiT

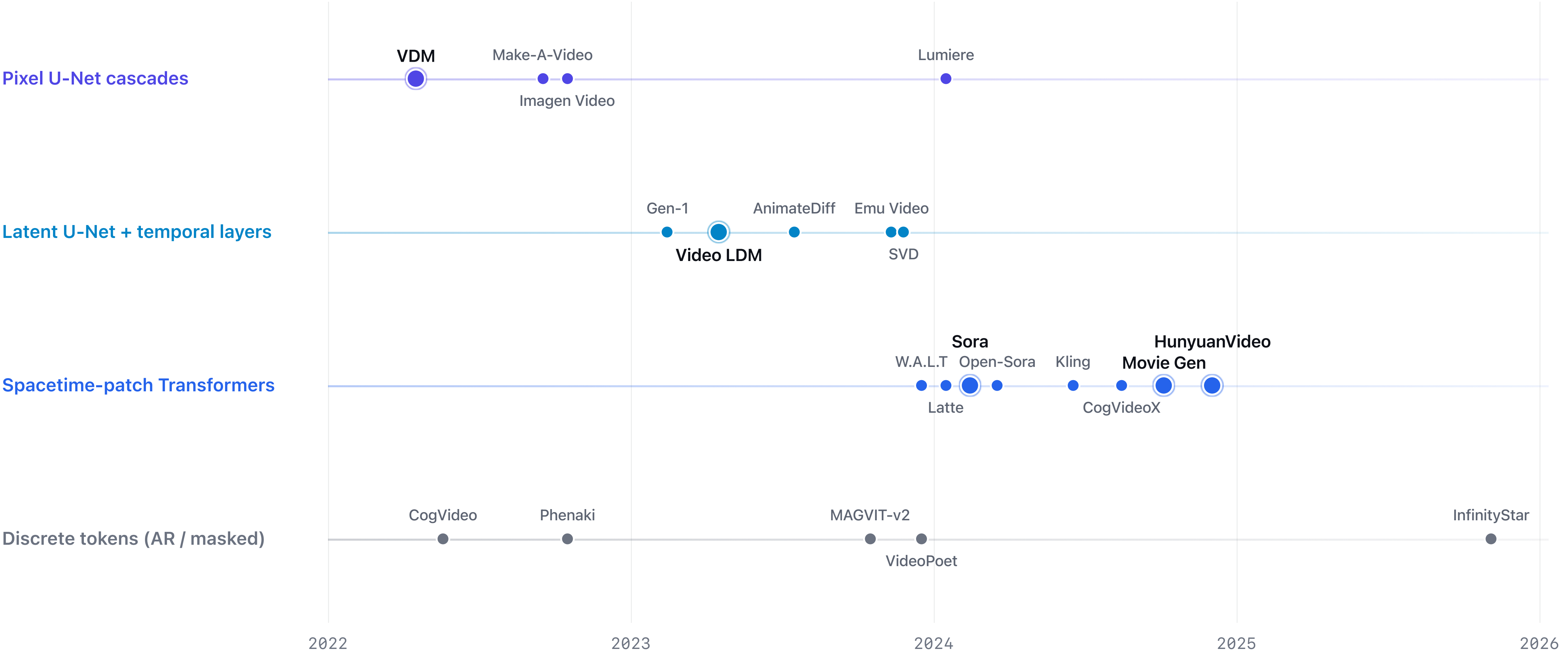
Movie Gen · 30B

HunyuanVideo · open weights

Video: OpenAI, Sora (2024) · generated, not a drone shot

07

Video diffusion changed architecture **twice in two years**

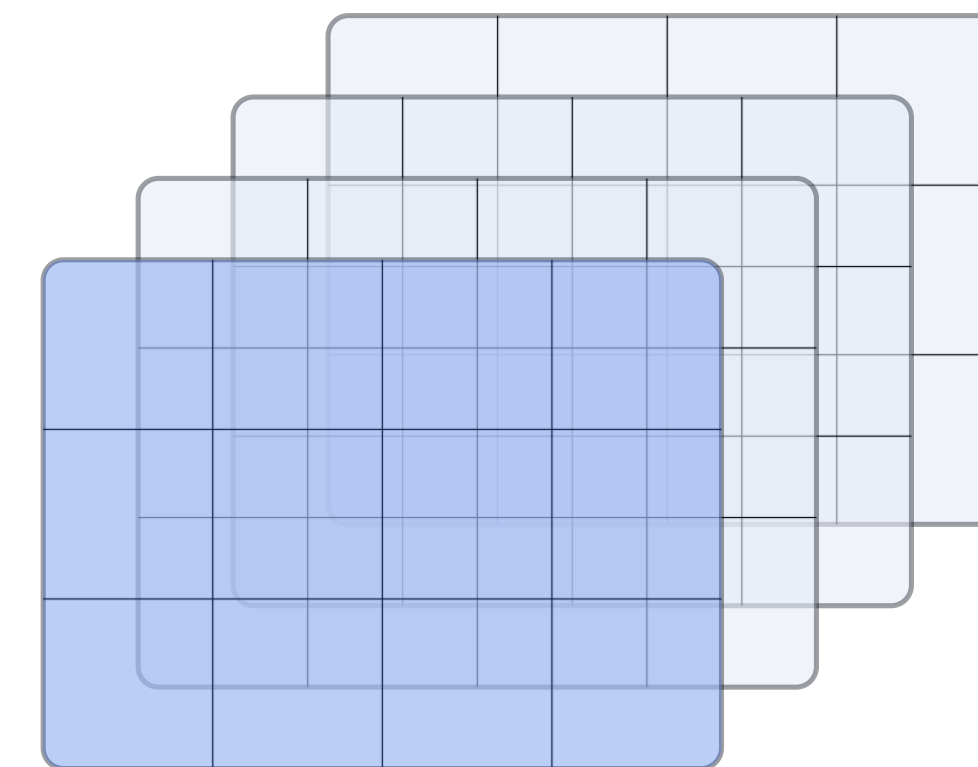


→ Each switch answered one question: how to afford the extra time axis.

VDM: inflate the image U-Net, **factorize space and time**

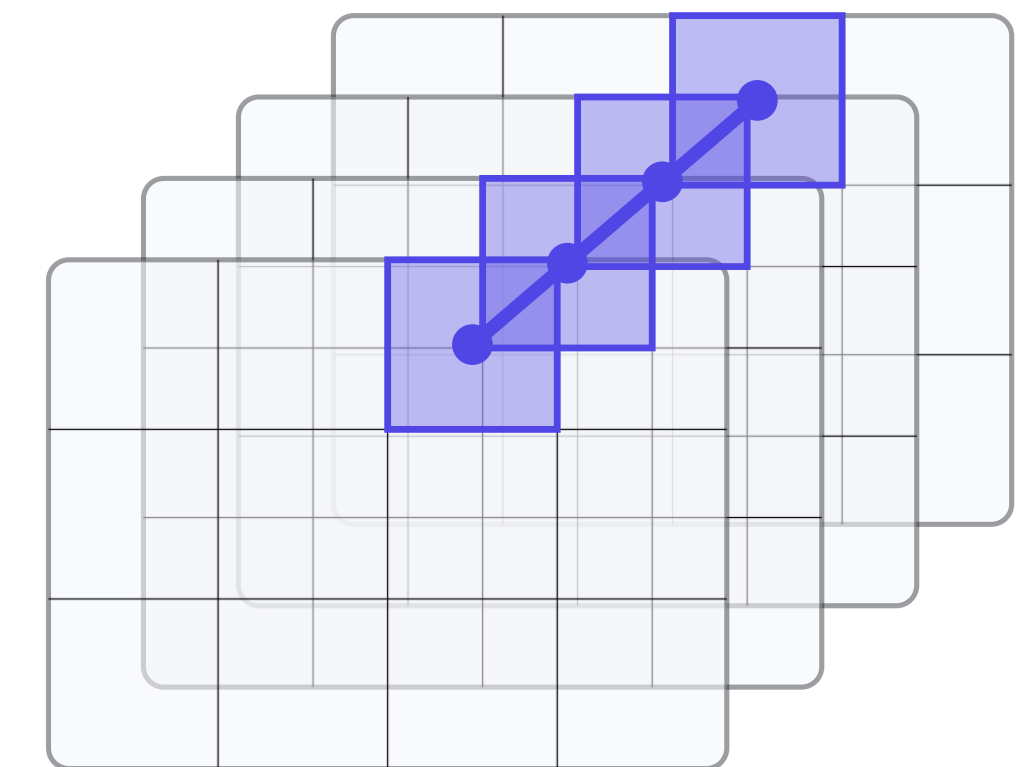


Video: Ho et al., VDM project page (2022) · composite, captions added



spatial attention

frames → batch



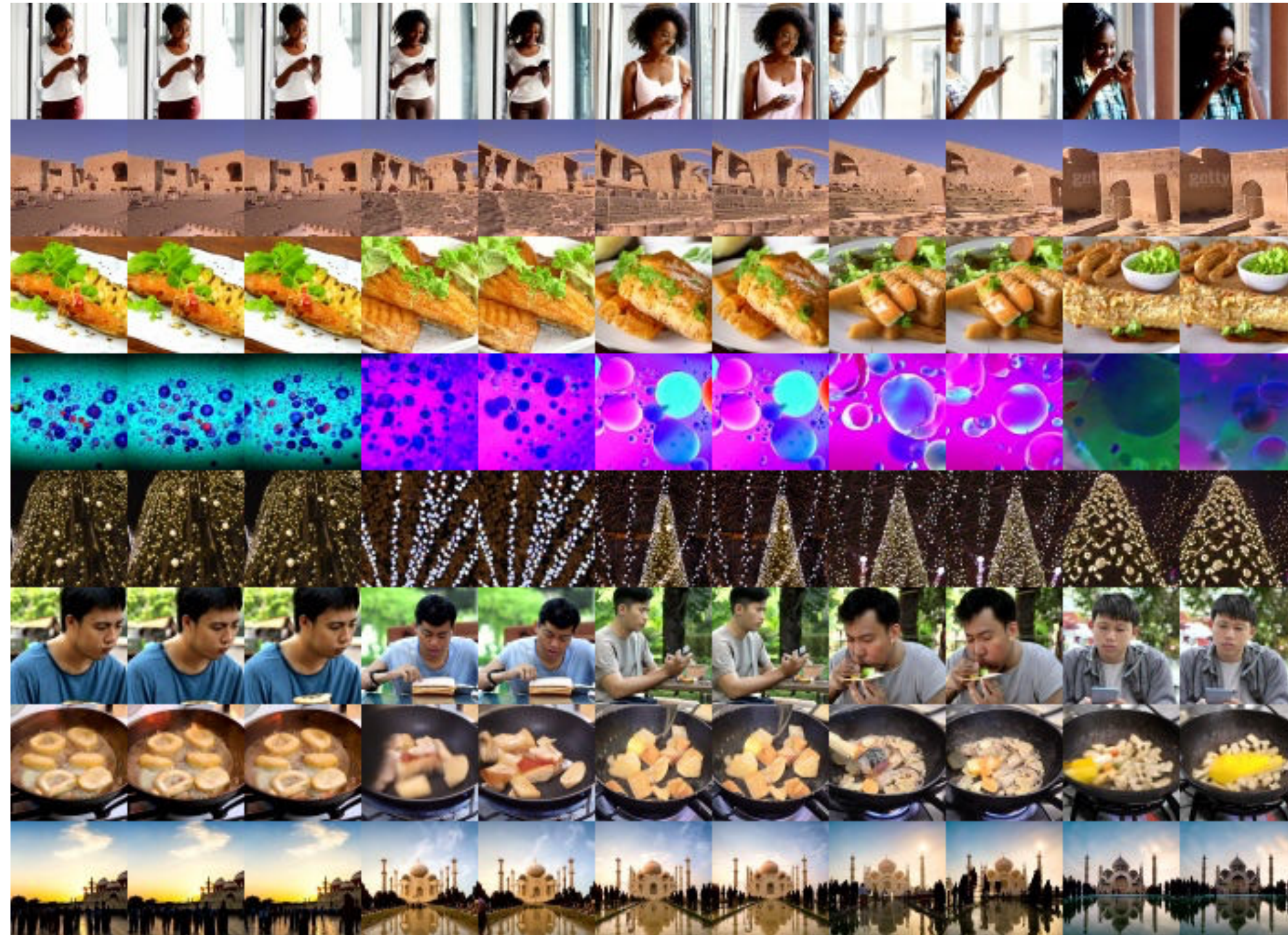
temporal attention

positions → batch

Joint image–video training: extra images ride along as frames; their temporal attention is masked to identity.

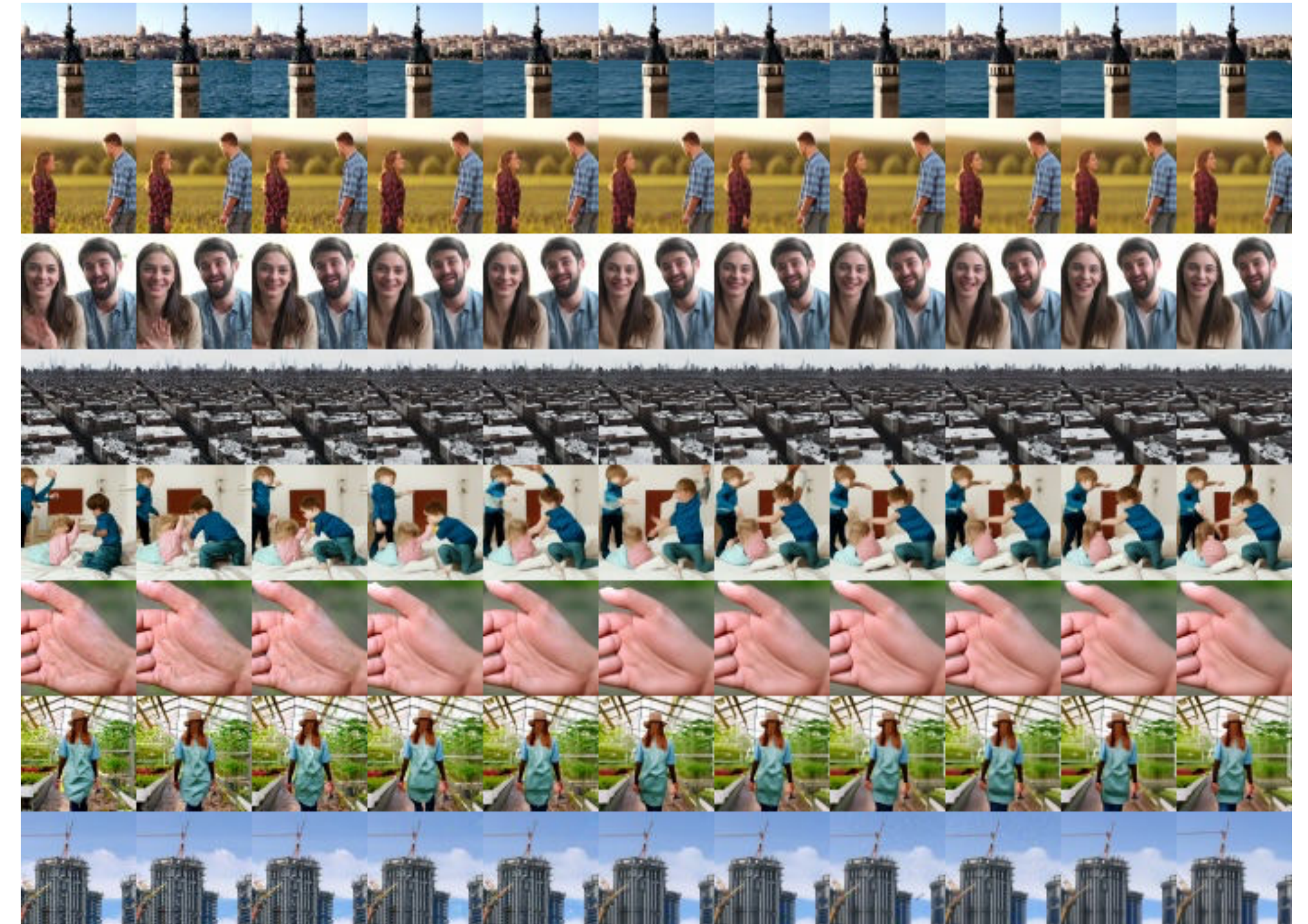
→ Two VDM defaults: separate space/time layers (until Sora-era full attention) and joint image-video training.

Extending a clip needs **reconstruction guidance**



Replacement: new frames drift from the given ones

Ho et al., 2022 · Fig. 4 left (adapted)



Reconstruction guidance: coherent with them

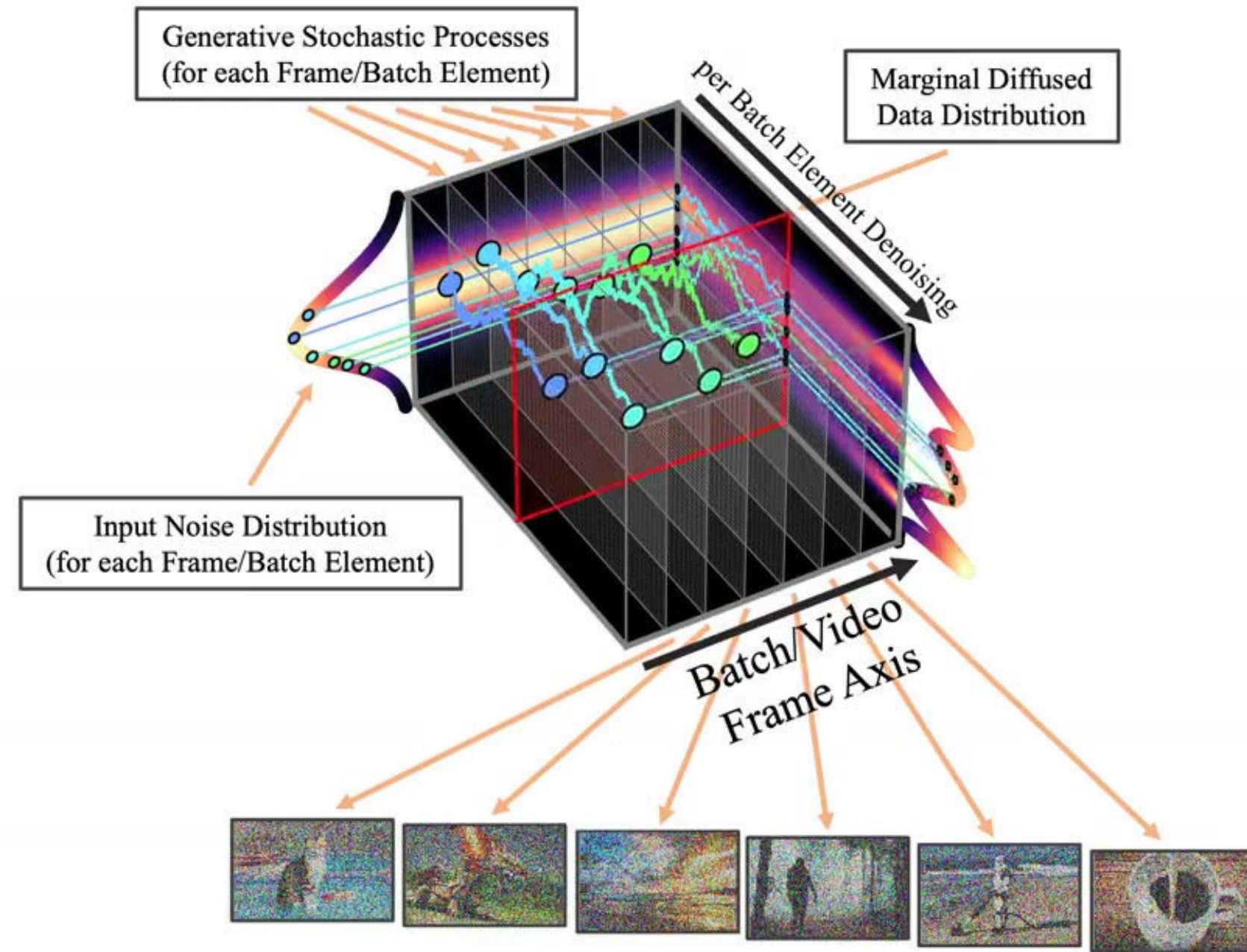
Ho et al., 2022 · Fig. 4 right (adapted)

EACH STEP: CORRECT THE PREDICTION FOR NEW FRAMES b SO THE RECONSTRUCTION OF KNOWN FRAMES a MATCHES THEM

$$\tilde{x}^b = \hat{x}^b - \frac{w_r \alpha_t}{2} \nabla_{z_t^b} \|x^a - \hat{x}^a(z_t)\|^2, \quad w_r > 1$$

→ Pasting in known frames fails; steering the model's own reconstruction toward them gives coherent extensions.

Freeze Stable Diffusion, **train only temporal layers**



Video: Blattmann et al., Video LDM project page (NVIDIA, 2023)



Video: Blattmann et al., Video LDM project page (NVIDIA, 2023) · 2x2 composite

LEARNED MIX WITH THE FROZEN IMAGE LAYER

$$z \leftarrow \alpha z + (1 - \alpha) l_{\phi}(z)$$

l_{ϕ} : new temporal layer · α learned; $\alpha = 1$ is the image model

512×1024 driving

T2V up to 1280×2048

[transfers to DreamBooth models](#)

→ Images teach appearance, scarce video teaches only motion: the 2023 template behind AnimateDiff and SVD.

Sora: minute-long video from a diffusion transformer



~1 min

LONGEST VIDEO

2048²

IMAGES, SAME MODEL

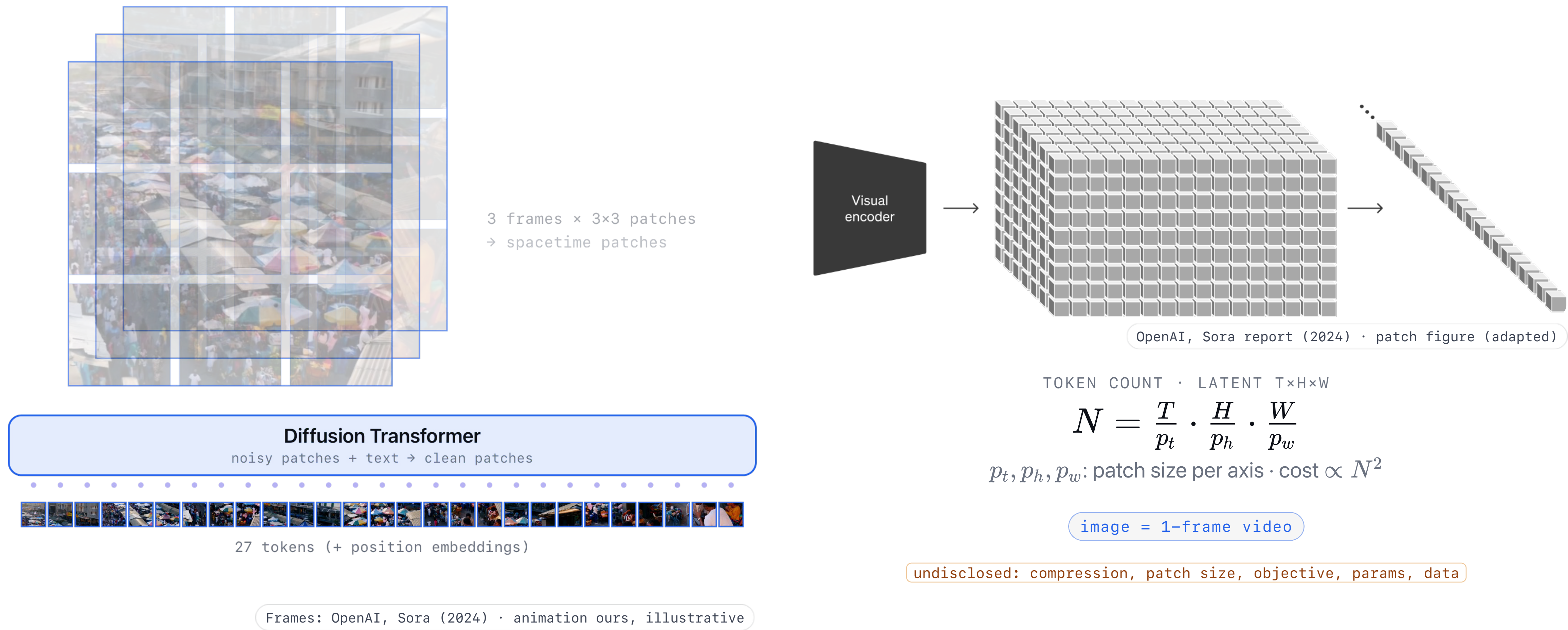
1920×1080 ↔ 1080×1920

NATIVE ASPECT RATIOS

objective · size · data · compute undisclosed

→ Known parts, combined and trained at scale, gave minute-long coherent video in 2024; the details stayed secret.

Video as one sequence of **spacetime patches**



Real patches are latent, not pixels.

→ Patches turn any size, length or aspect ratio into one token sequence for a diffusion Transformer.

Train at **native size**, sample any shape



trained on square crops

native aspect

base compute

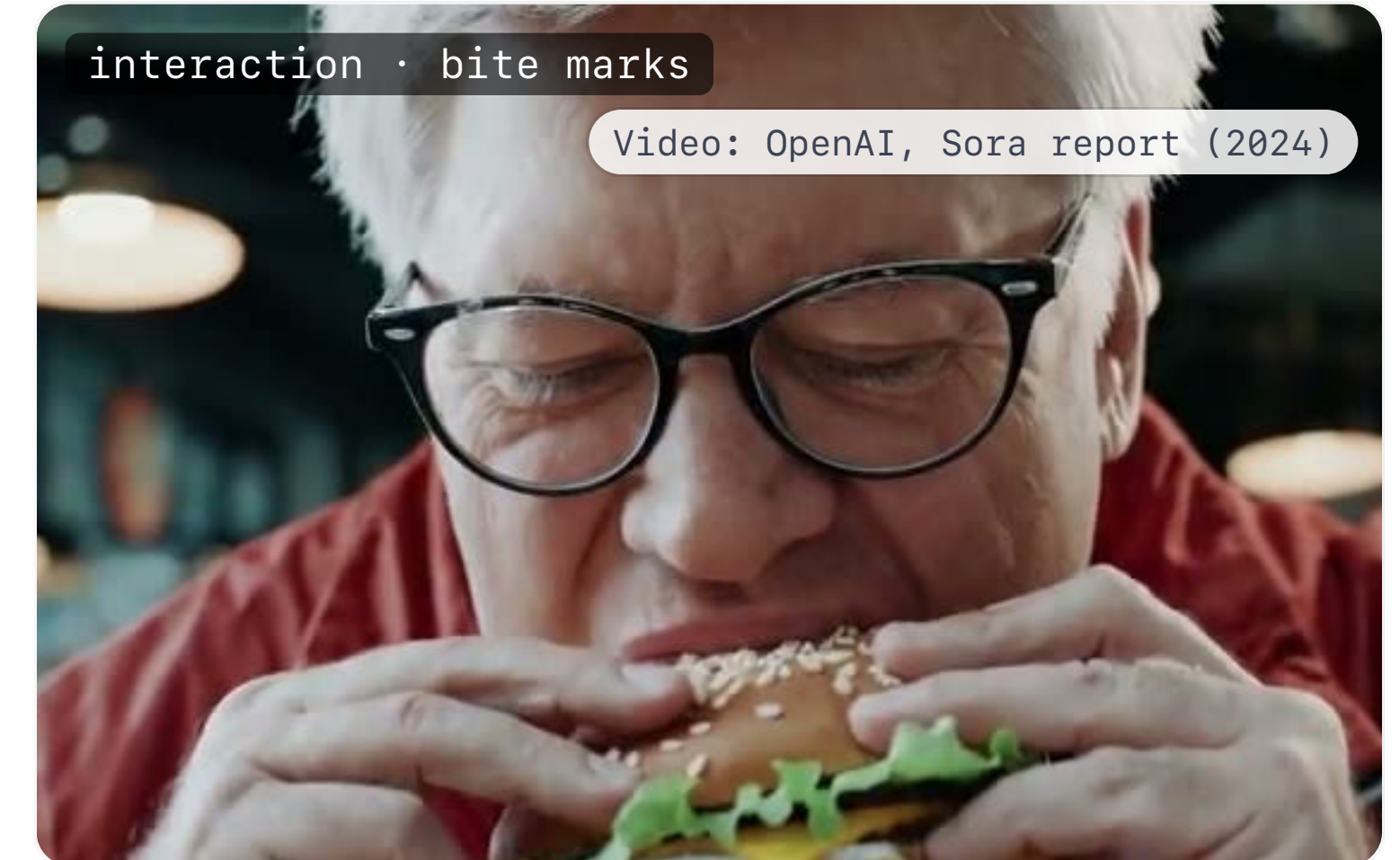
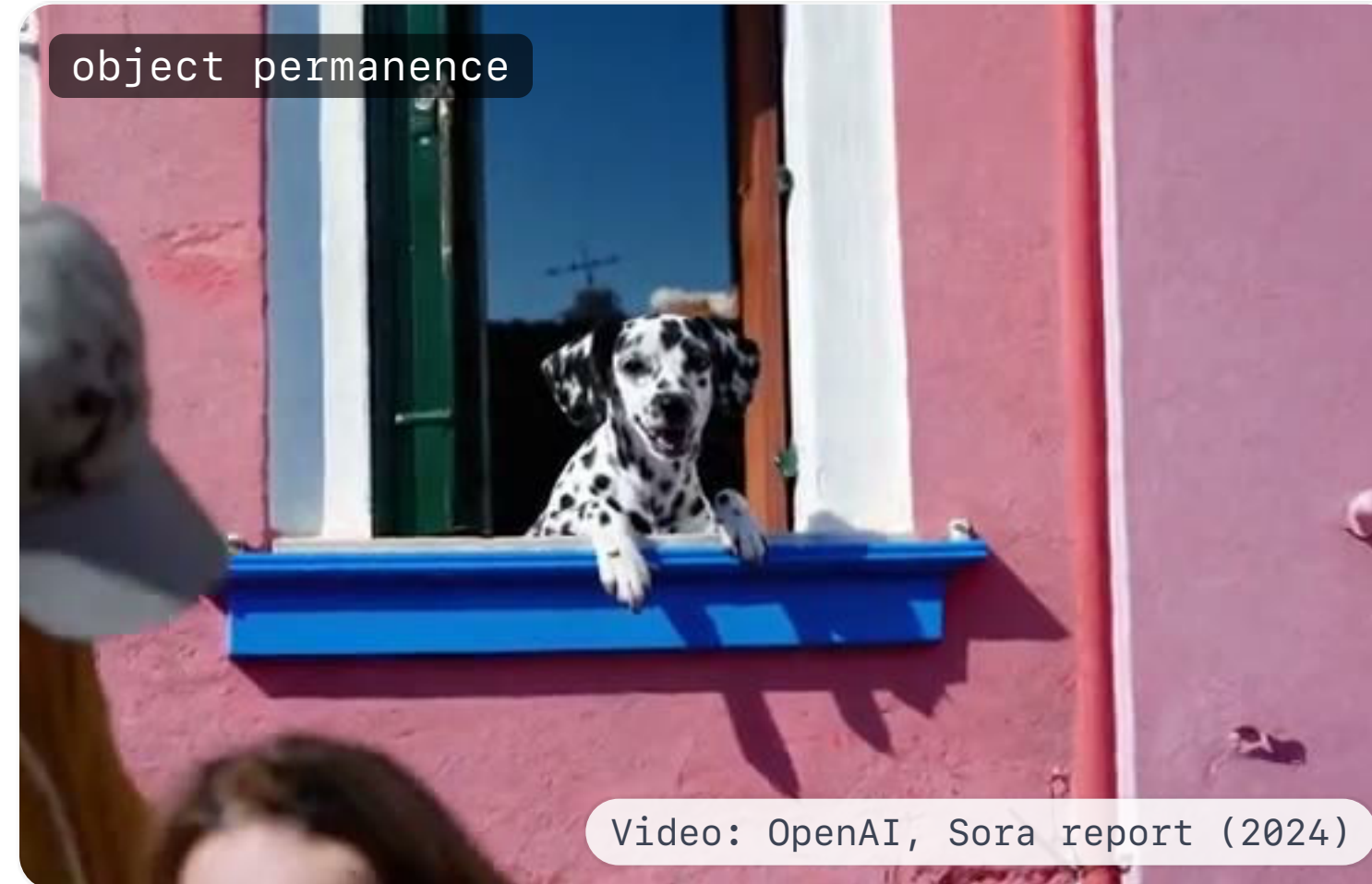
4x

16x

mixed-size batching undisclosed

➔ No cropping: native aspect ratios frame subjects better, and a fixed seed keeps improving with compute.

“World simulator”: emergent, but brittle

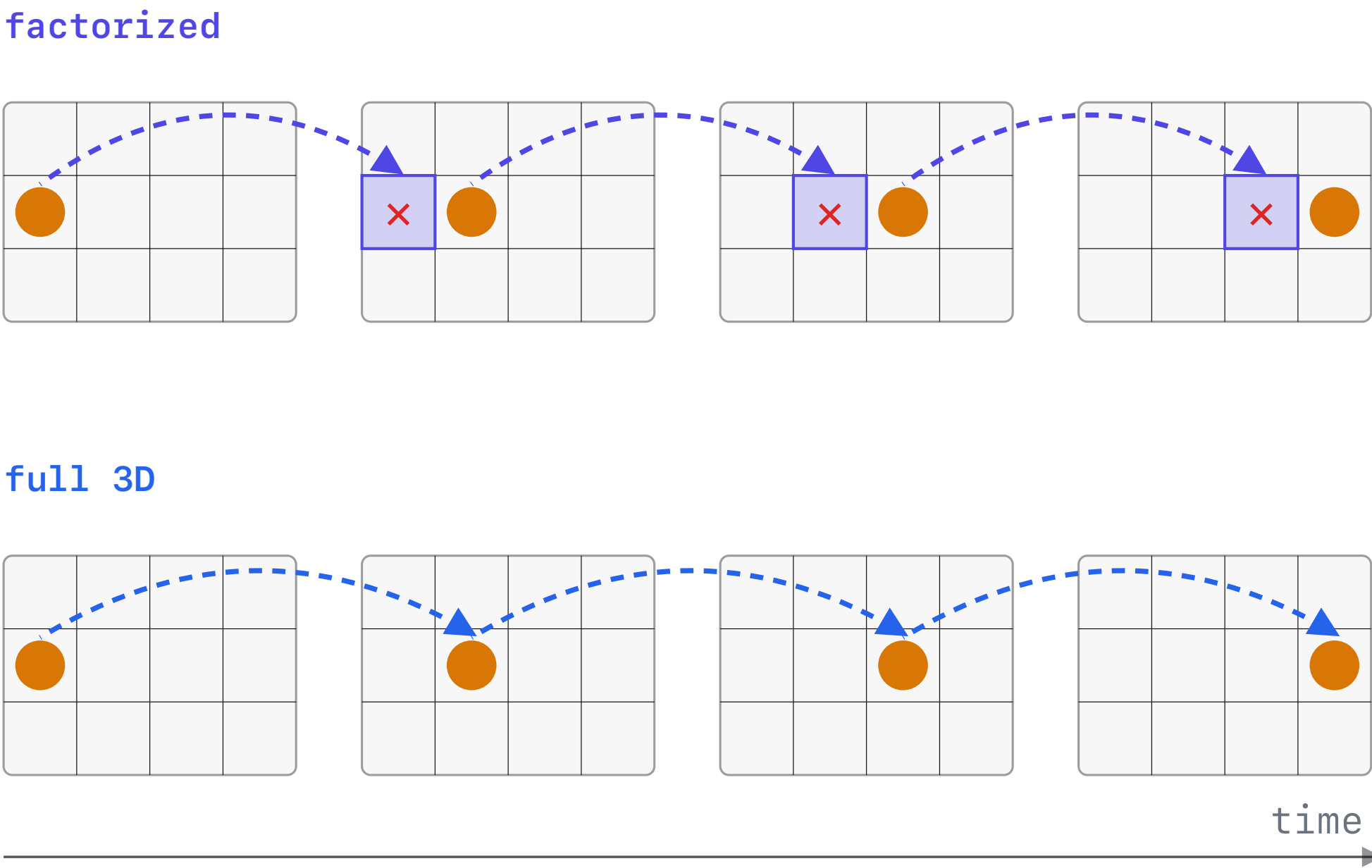
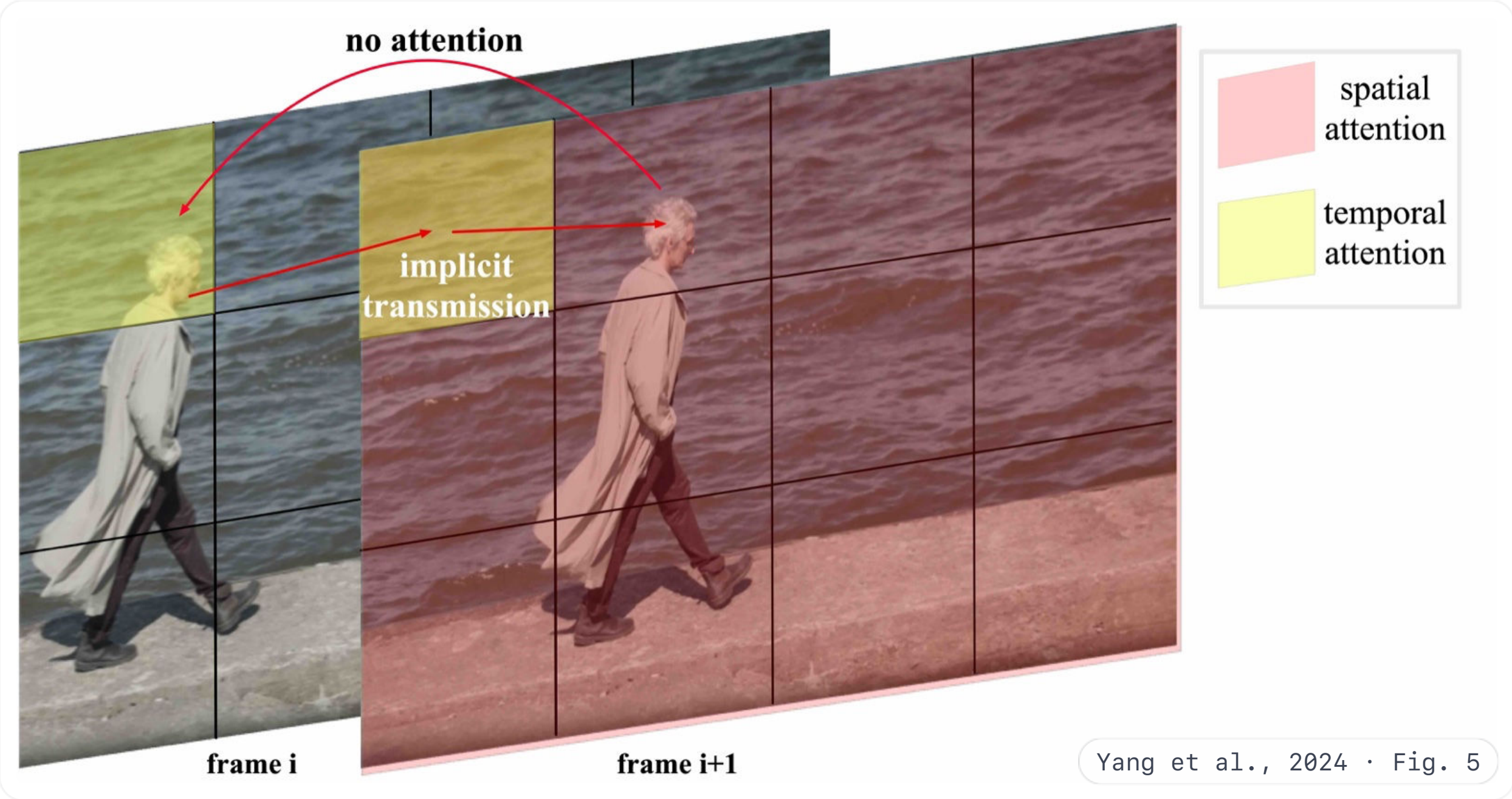


“scaling video generation models is a **promising path** towards building general purpose simulators of the physical world”

qualitative claims · no metrics

→ Scale bought surprising consistency, but the report's own failures show physics is not reliably learned.

Factorized attention misses large motion



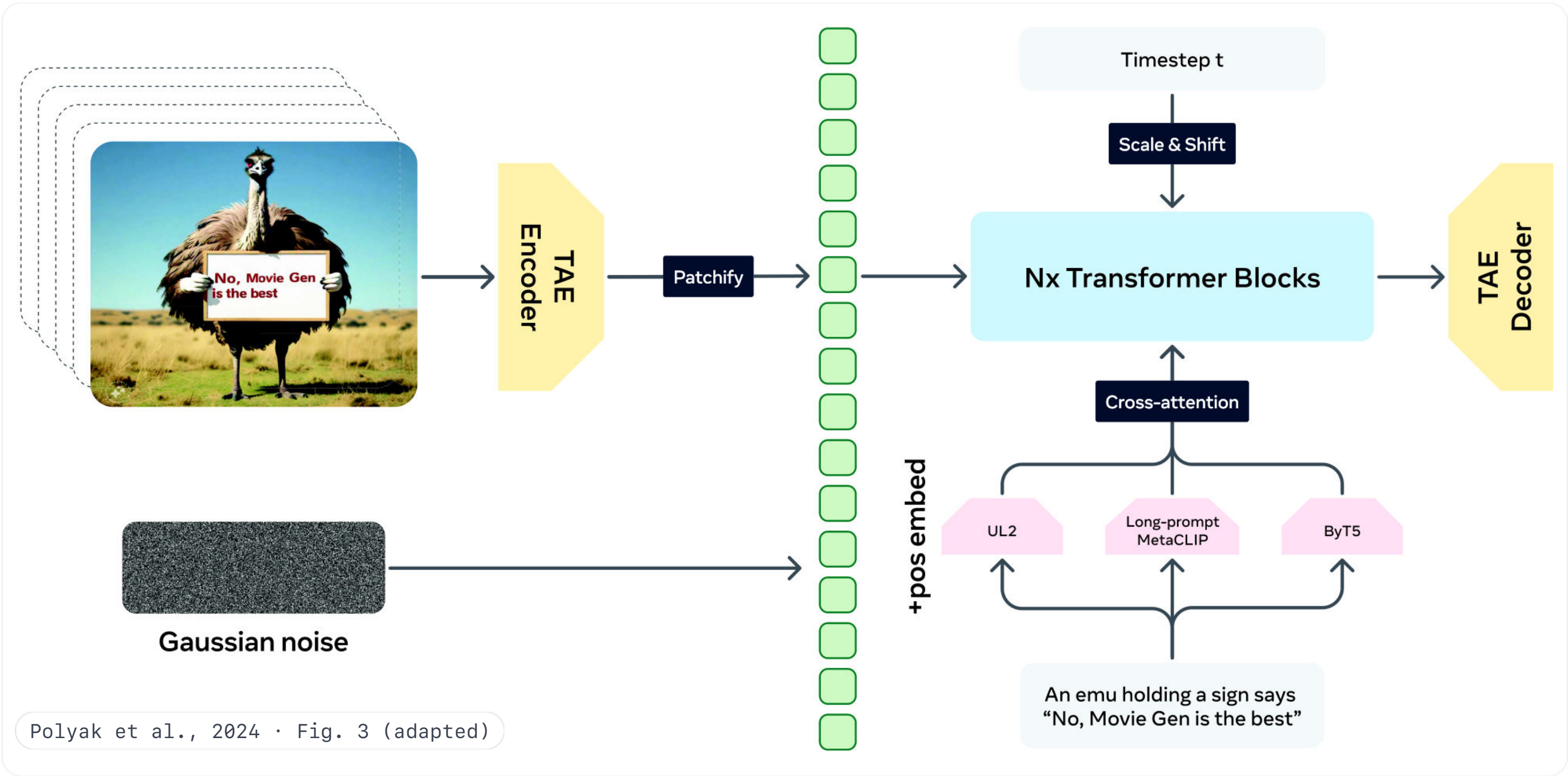
ATTENTION COST PER LAYER

$$\underbrace{T(HW)^2 + HW T^2}_{\text{factorized}} \ll \underbrace{(THW)^2}_{\text{full}}$$

~73K tokens: Movie Gen, 16 s at 768², after a 3D VAE (compresses time too) + patchify

→ Full attention became affordable once 3D VAEs and patches cut clips to tens of thousands of tokens.

Movie Gen: LLaMa3 blocks + flow matching at 30B



text encoders: UL2 · long-prompt MetaCLIP · ByT5



Video: Meta, Movie Gen (2024)

FRAMES × HEIGHT × WIDTH TOKENS

$$\frac{256}{8} \times \frac{768}{16} \times \frac{768}{16} = 73,728$$

16 s · 768² · TAE 8×8×8 · patch 1×2×2

flow matching > v-pred + zero SNR
(their ablation)

→ 1080p upsampler

Audio 13B · 48 kHz

no weights

→ The most complete recipe a frontier lab published in 2024: choices ablated, written down, LLM tools reused.

HunyuanVideo: the recipe with open weights



Video: Tencent, HunyuanVideo (2024) · cropped

13B+

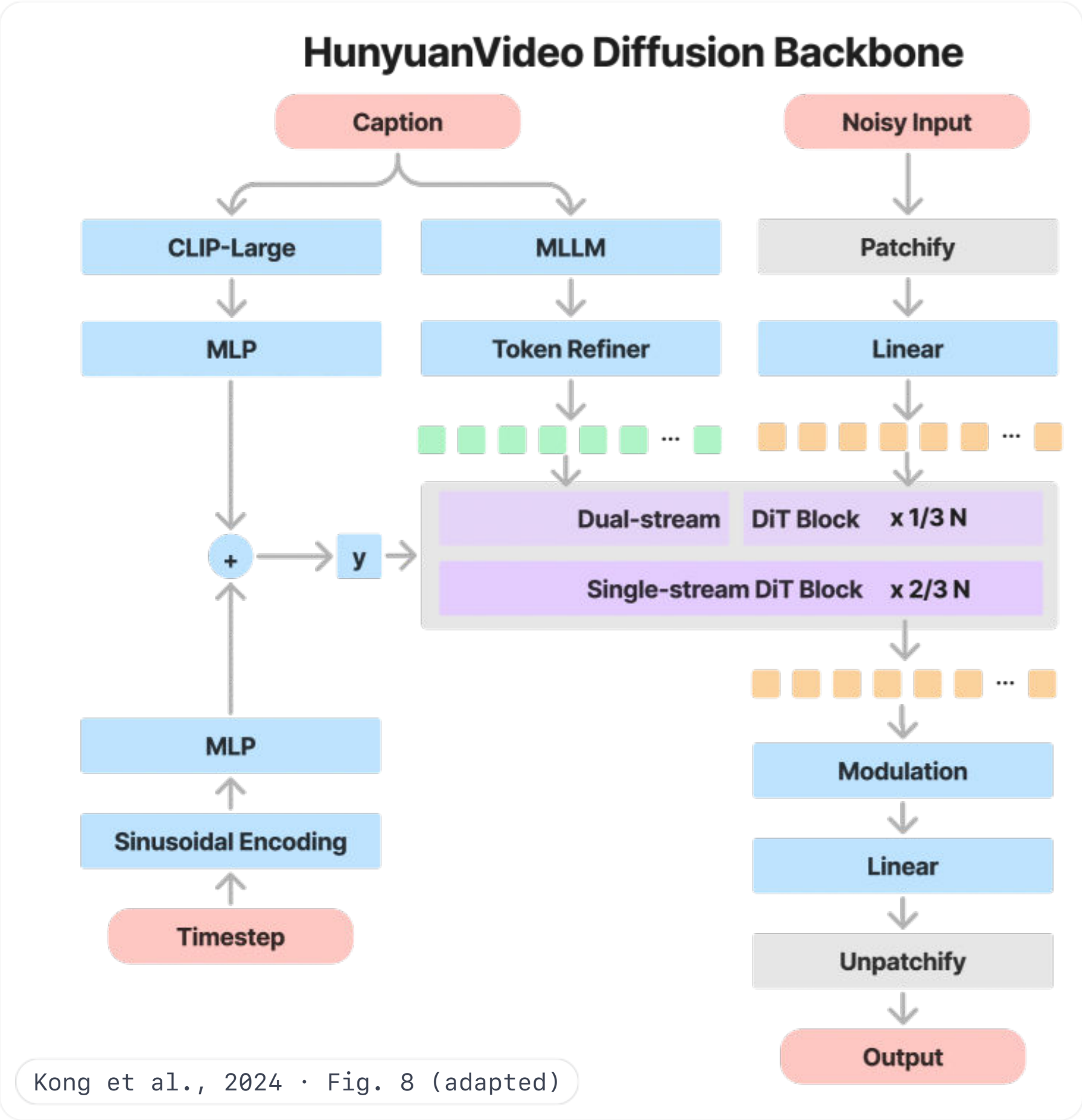
LARGEST OPEN VIDEO MODEL THEN

4x8x8

CAUSAL 3D VAE · 16 CH

- dual → single stream
- MLLM text encoder + CLIP-L
- flow matching · 3D RoPE

→ Near-frontier quality with open weights made the 2024 recipe a shared research platform.



Five choices, **one converged recipe**

Latent	pixels → per-frame 2D VAE → 3D VAE · causal 4×8×8 / TAE 8×8×8	VDM → CogVideoX
Backbone	factorized 3D U-Net → frozen SD + temporal layers → full-attention Transformer	VDM → CogVideoX
Objective	ϵ (some v) → SD objective → v-pred + zero SNR → flow matching	VDM → Movie Gen
Text encoder	BERT → CLIP → T5 → UL2 + ByT5 + MetaCLIP · MLLM + CLIP-L	VDM → HunyuanVideo
Captions	paired captions → recaptioning + GPT prompt expansion → VLM captioners · LLaMa3-Video	Sora → Movie Gen

➔ By end-2024 video diffusion meant 3D VAE + full-attention Transformer + flow matching + LLM-grade text.

— CHAPTER 8 · 2024 → 2026

Video becomes real-time, audible, interactive

Wan · open base

Diffusion Forcing

CausVid → Self Forcing

Veo 3 · native audio

Genie 3 · world models

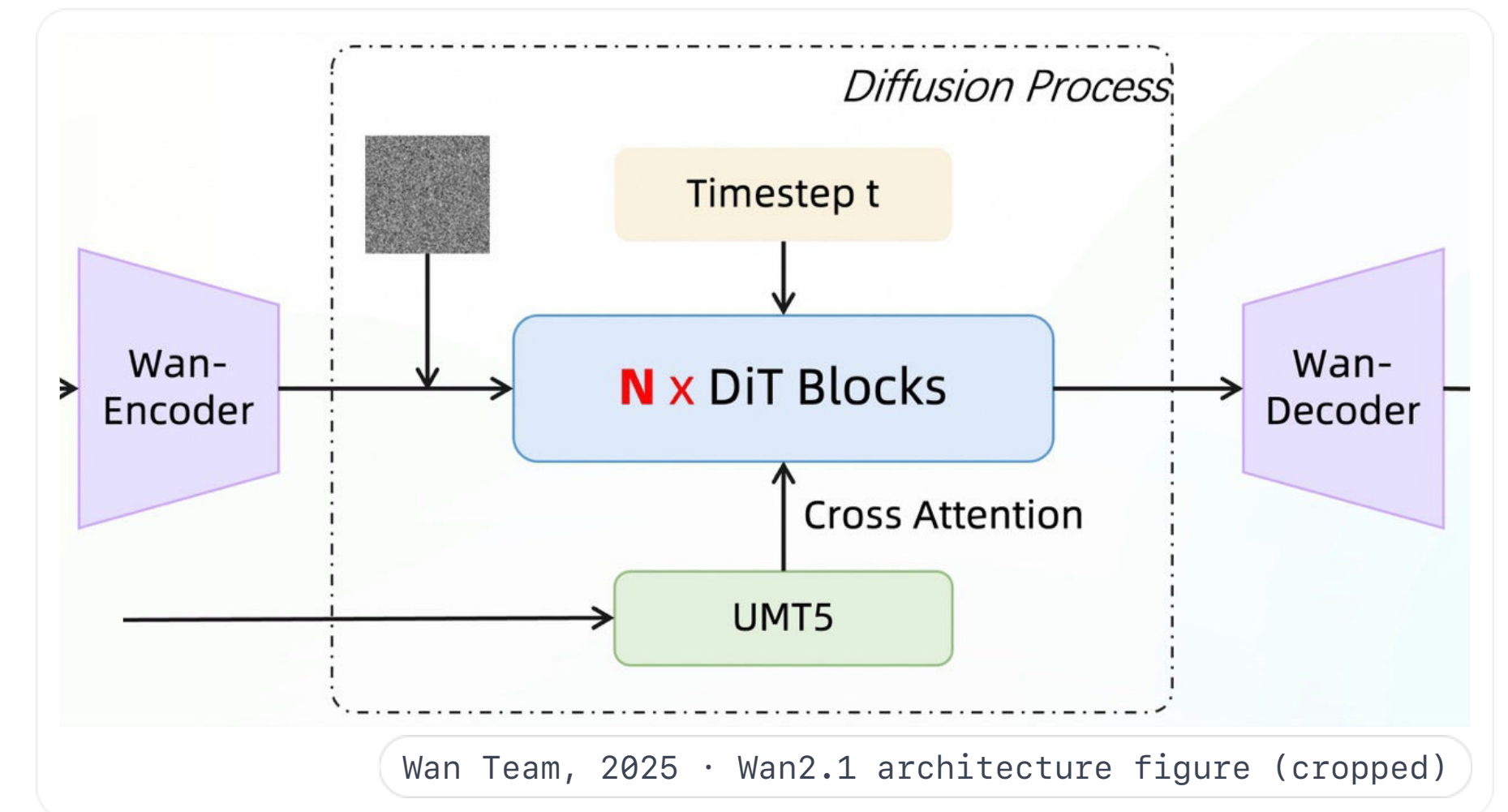
zero-shot reasoners

omni models

Video: Google DeepMind, Genie 3 (2025) · generated, playable world

08

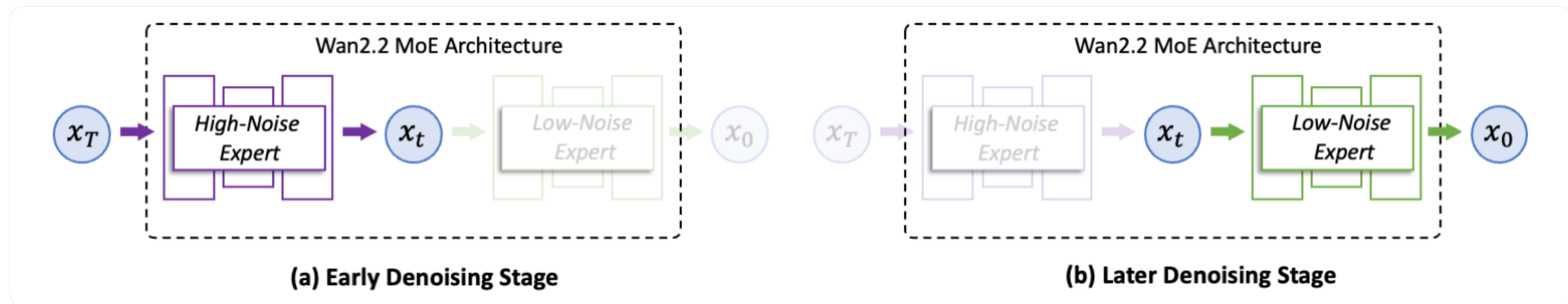
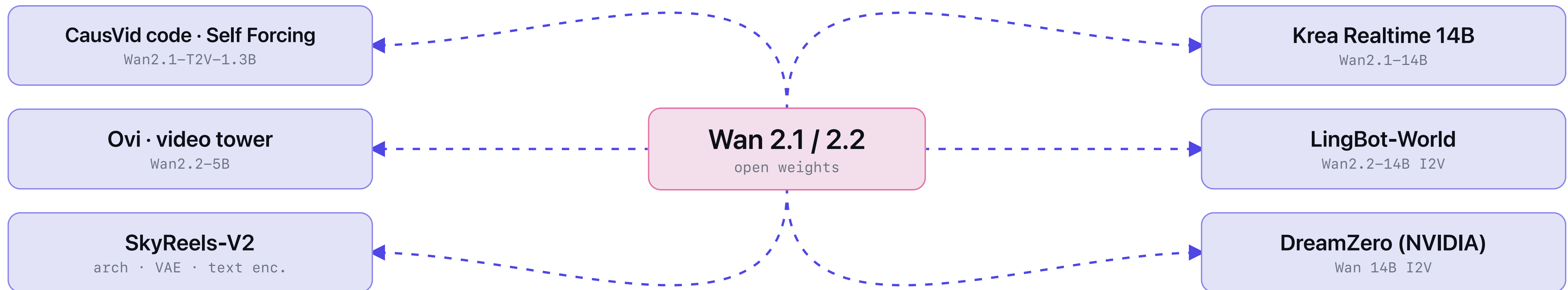
Wan made a strong video base **fully open**



- **Open:** Apache 2.0 weights + code
- **Wan-VAE:** causal 3D, $4 \times 8 \times 8$ compression
- **1.3B** needs ~8.19 GB VRAM: 5 s of 480p in ~4 min on an RTX 4090

→ Strong, free and runnable on one gaming GPU: arguably the shared starting point for video research.

Most open work in this chapter is **built on Wan**



Wan Team, 2025 · Wan2.2 MoE figure

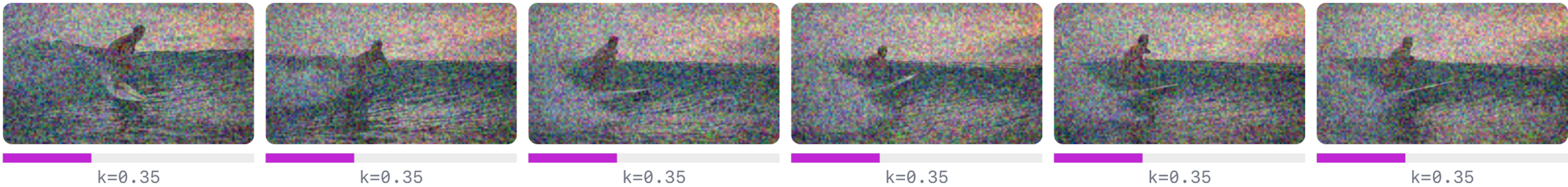
Wan 2.2: expert v_{high} while $t \geq t_{\text{moe}}$ (larger t = noisier), then v_{low} · **27B** total, **14B** active per step

→ One reading: research moved from “train a base first” to “post-train an open base”, as with Stable Diffusion.

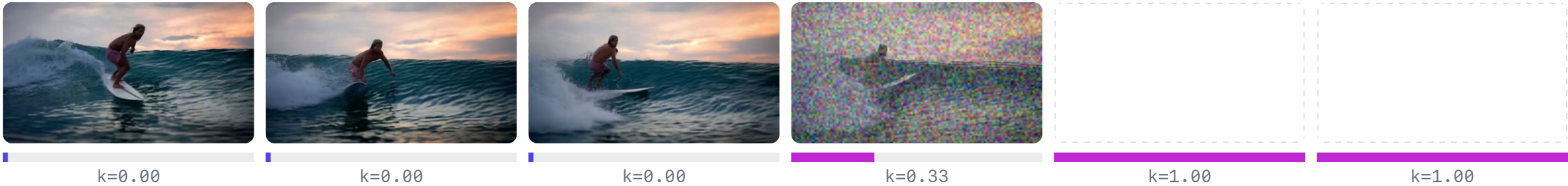
Give every frame **its own noise level**

Frames: Diffusion Forcing project page · animation ours

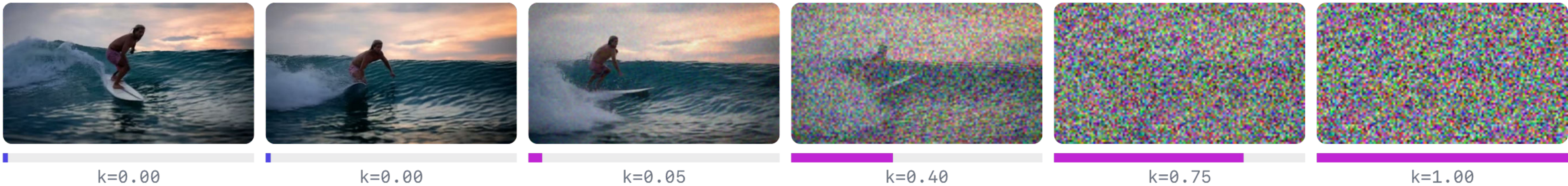
Full-sequence diffusion
all frames share one noise level



Autoregressive (next frame)
past frames clean, one frame at a time



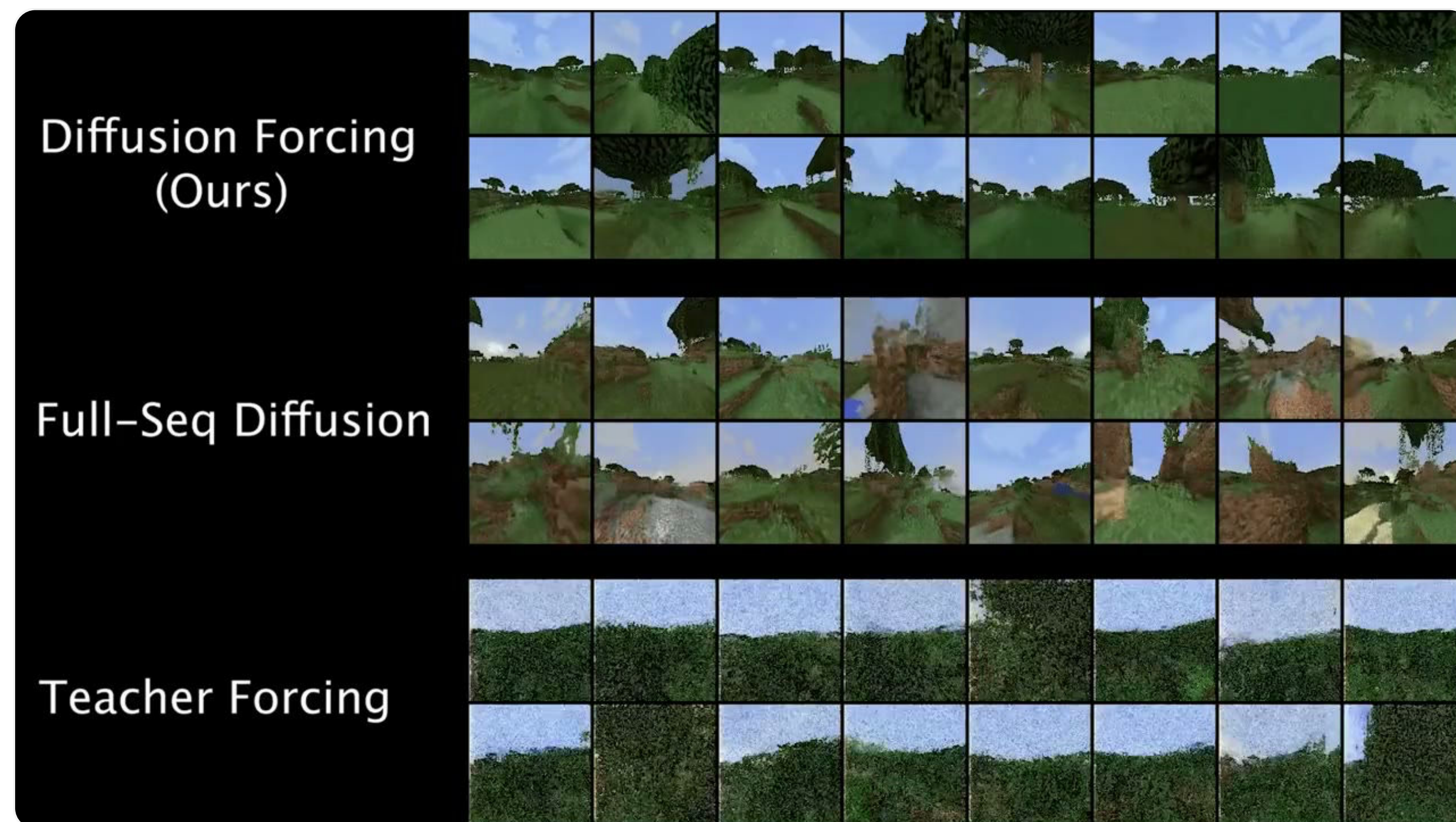
Diffusion Forcing
independent per-frame noise levels



frame 1 $\rightarrow$ 6 · bar = noise level k (1 = pure noise)

→ Next-frame prediction and full-sequence diffusion are two noise patterns; Diffusion Forcing trains on all of them.

Slightly noisy history keeps rollouts stable



Video: Diffusion Forcing project page · Minecraft rollouts

ONE NOISE LEVEL PER FRAME

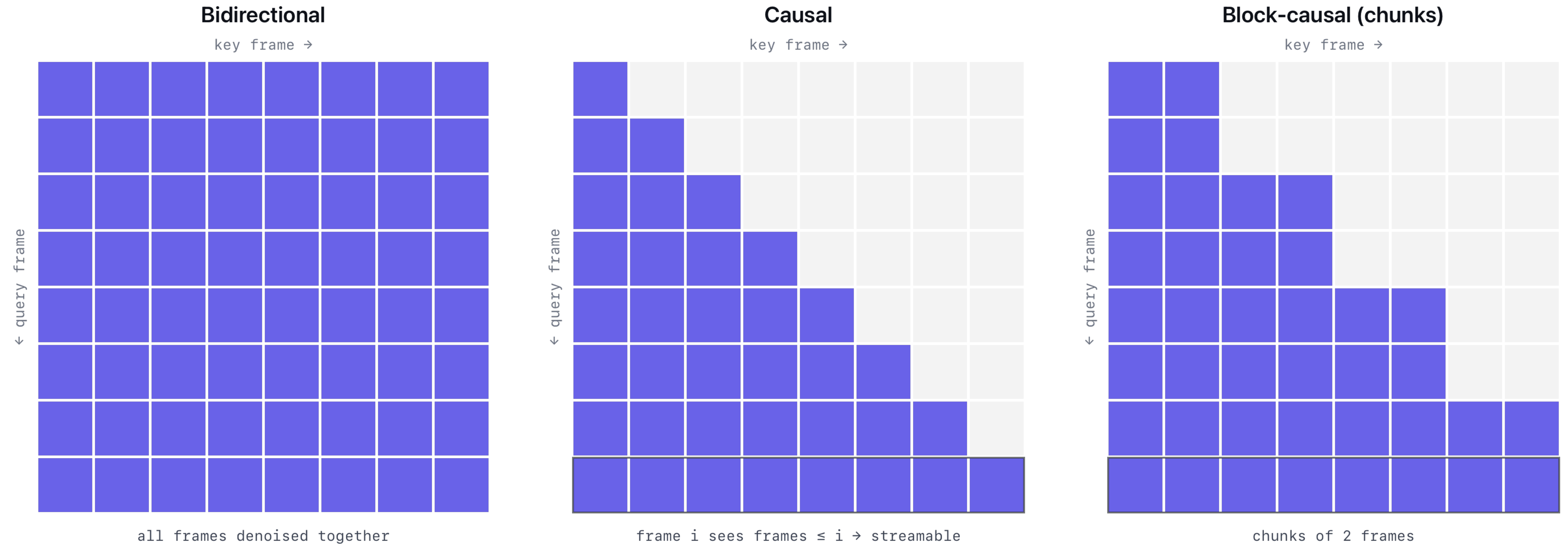
$$\mathcal{L} = \mathbb{E}_{k_t \sim \mathcal{U}\{0..K\} \text{ iid}} \sum_t \|\epsilon_t - \epsilon_\theta(x_1^{k_1}, \dots, x_t^{k_t}; k_{1:t})\|^2$$

Frame t is noised to its own random level k_t ; a causal net denoises every mix and is told every level $k_{1:t}$.

Teacher forcing: clean past, noisy present. Full-sequence: one shared level.

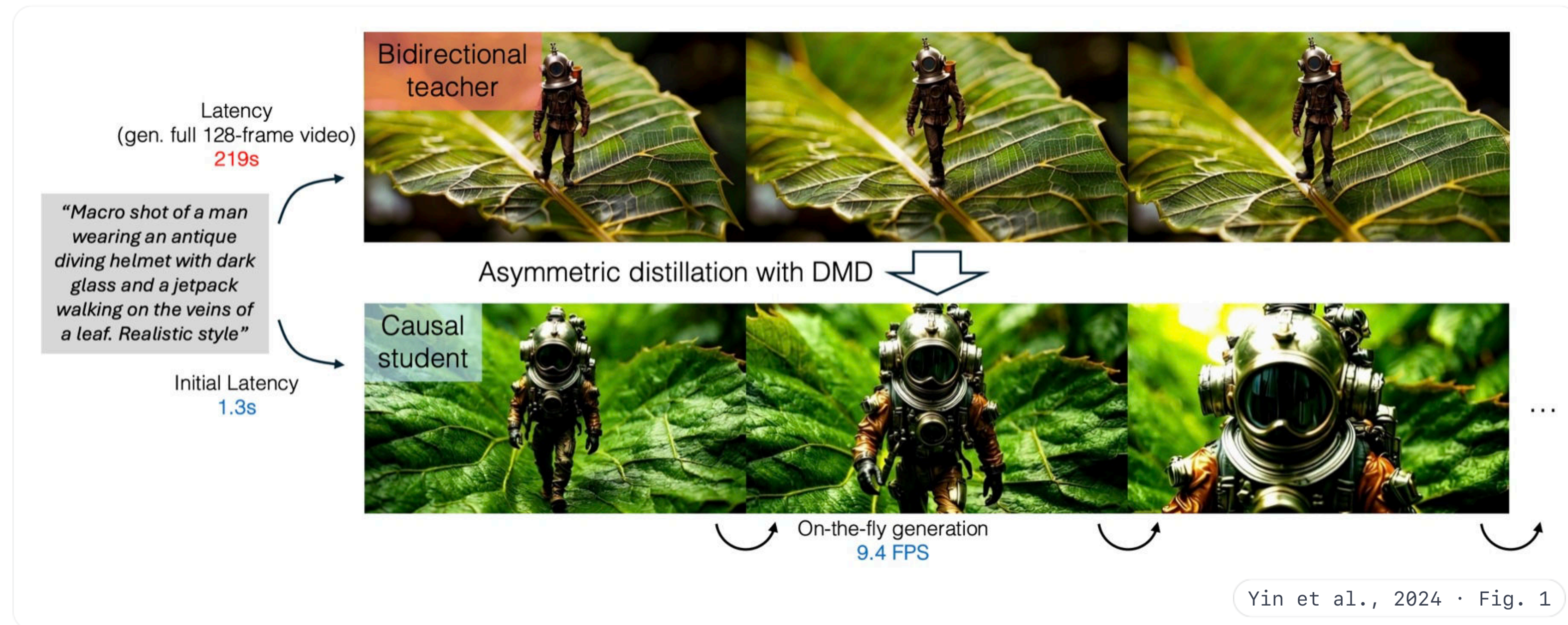
→ Train the model to expect an imperfect past, and its own mistakes stop snowballing.

To stream video, a frame may only **look back**



→ A bidirectional model shows nothing until every frame is done; a causal one can show frames as they come.

A bidirectional teacher trains a **fast causal student**



219 s → 1.3 s

TEACHER, 128 FRAMES → STUDENT'S FIRST FRAME

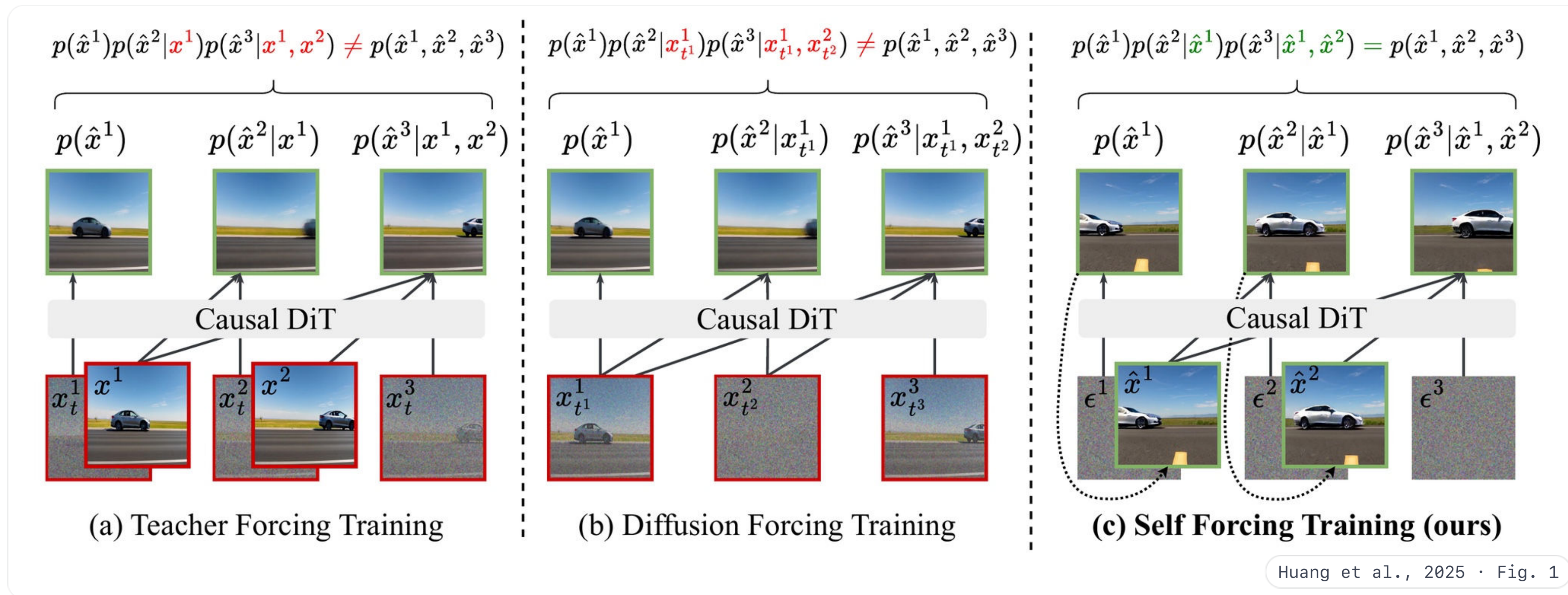
9.4 FPS

STREAMING AFTER THAT, ONE GPU

- block-causal attention + KV cache (past frames not recomputed)
- DMD: 50 → 4 steps
- student initialized on teacher ODE pairs

→ Asymmetric distillation: a teacher that sees the future grades a student that never does.

Train on your own rollouts to close the train–test gap



VIDEO-LEVEL DISTRIBUTION MATCHING

$$\min_{\theta} \mathcal{D} \left(\prod_i p_{\theta}(x_i | \hat{x}_{<i}) \parallel p_{\text{data}}(x_{1:N}) \right)$$

via DMD; the teacher stands in for real data

(a) real past · (b) noisy real past · **(c) past = its own samples**, as at test time

→ Like learning to drive by driving yourself, with the instructor grading the whole trip, not each turn.

Real-time streaming **without losing quality**



Video: Self Forcing project page · 3 clips side by side (adapted)

17 FPS

ONE H100

0.45 s

LATENCY, FRAME-WISE

~150x

LOWER LATENCY THAN WAN2.1

~1.5 h

DMD TRAINING, 64 H100

Follow-ups: LongLive (240 s) · Self-Forcing++ (4 min 15 s) · Krea Realtime 14B

→ A post-training recipe: an open bidirectional base can cheaply be made causal, few-step and much faster.

Video generation leaves the **silent era**



“We’re emerging from the silent era of video generation.”

— Demis Hassabis, Google I/O 2025

TARGET

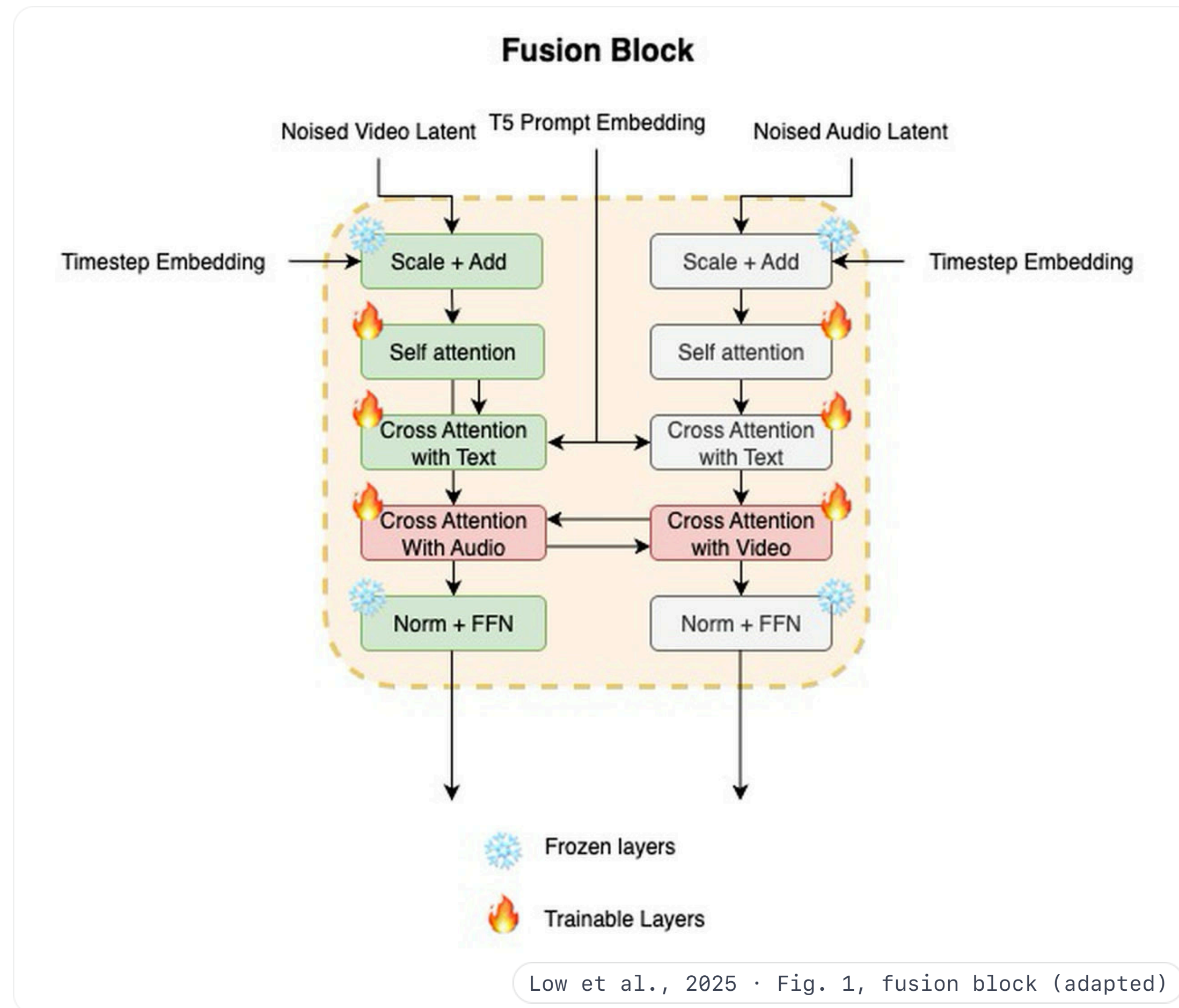
$$p(z^v \mid c) \longrightarrow p(z^v, z^a \mid c)$$

audio + video latents, diffused jointly by a transformer

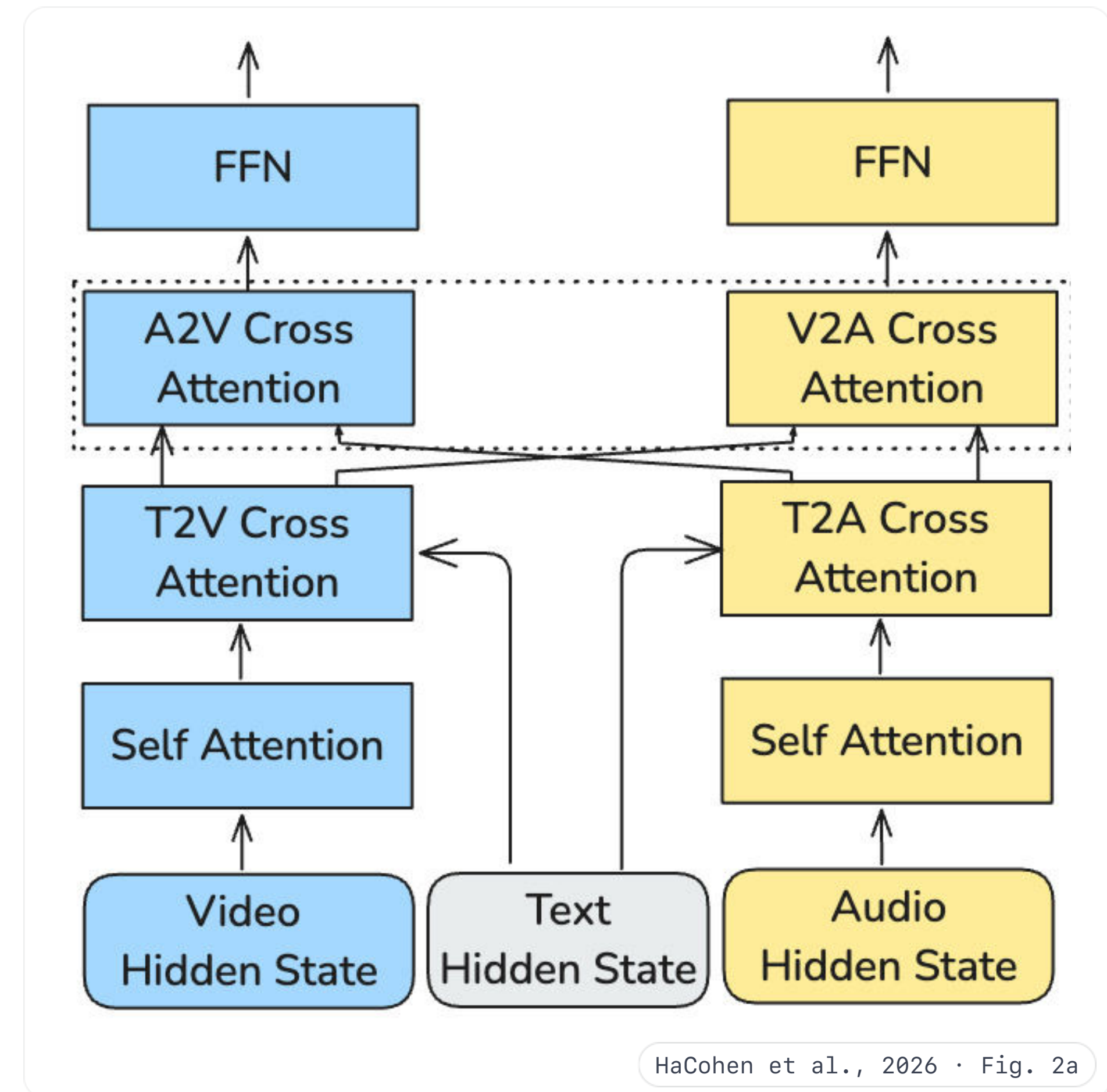
coupling & scale undisclosed

→ One prompt now yields picture, ambience, effects and lip-synced dialogue together, not dubbed later.

Open models show one way to fuse audio and video



Ovi: twin identical DiT towers, bidirectional cross-attention



LTX-2: 14B video + 5B audio streams, cross-attention

→ Two towers that talk through cross-attention: a public recipe, not a description of Veo 3.

A playable world, **generated frame by frame**



Video: Google DeepMind, Genie 3 (2025)

720p

24 FPS · REAL-TIME, FROM TEXT

minutes

MOSTLY CONSISTENT



Video: Google DeepMind, Genie 3 (2025)

~1 min

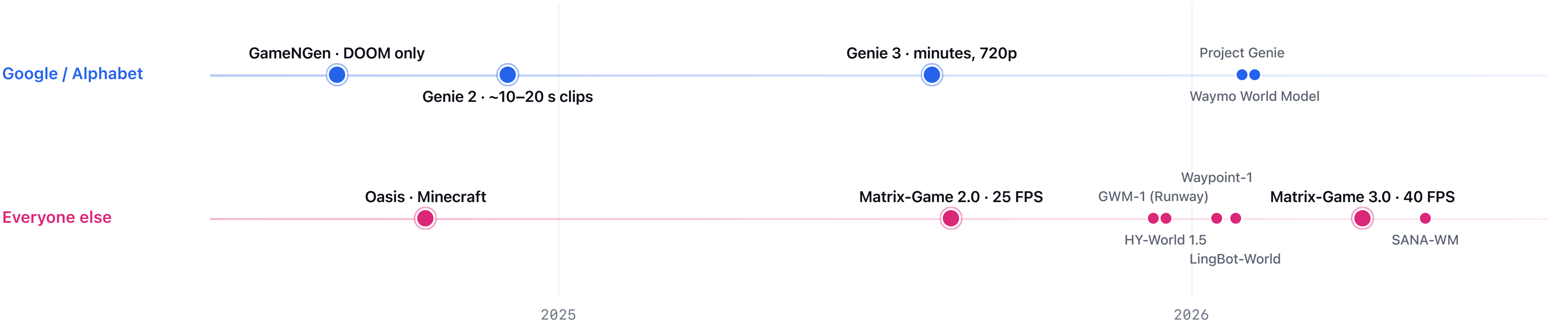
VISUAL MEMORY

architecture undisclosed

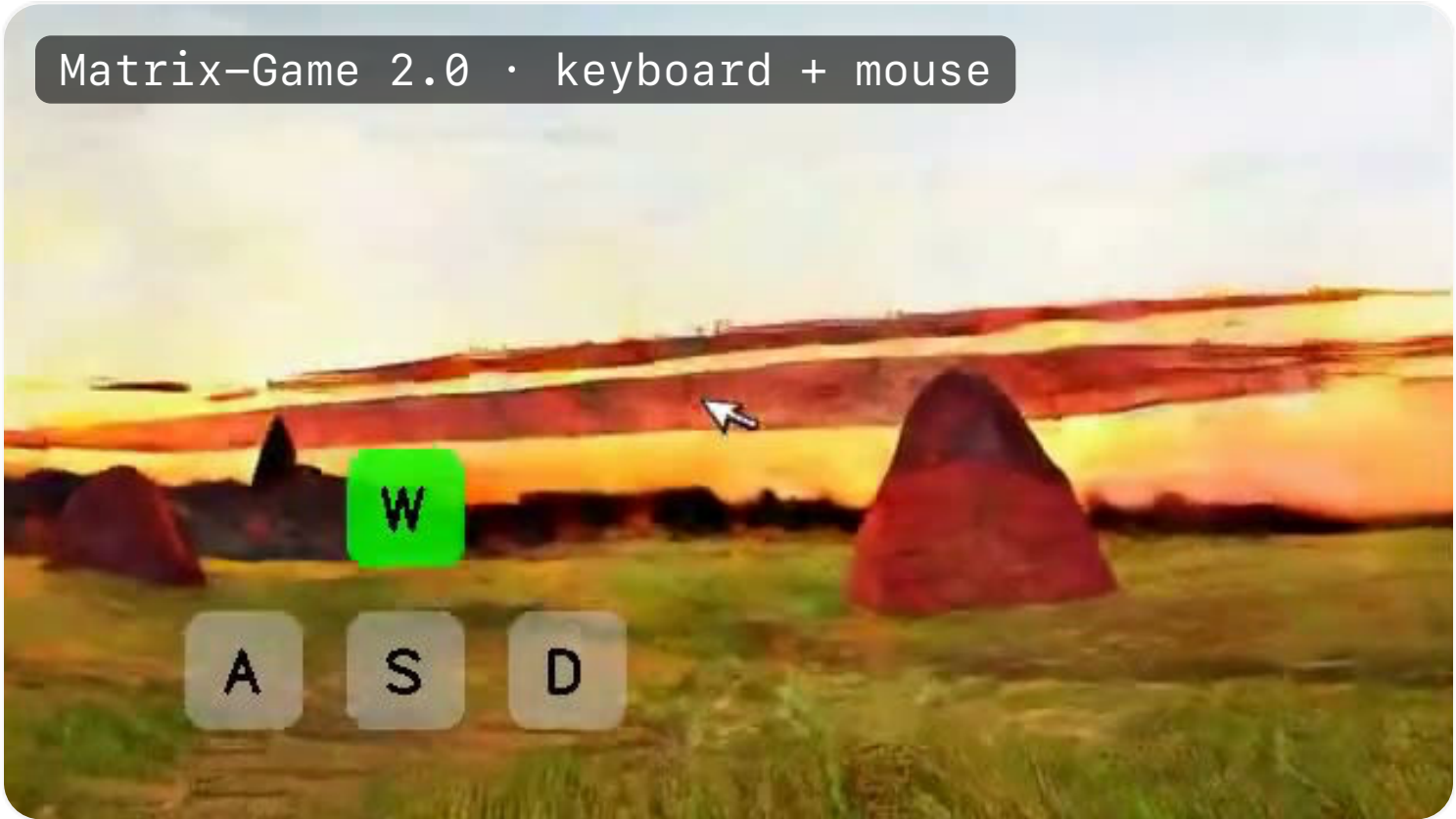
not stated to be diffusion

→ Arguably a game engine without a game engine: no stored geometry, each frame imagined from history + actions.

From one game to **any world**, in real time



Video: GameNGen project page



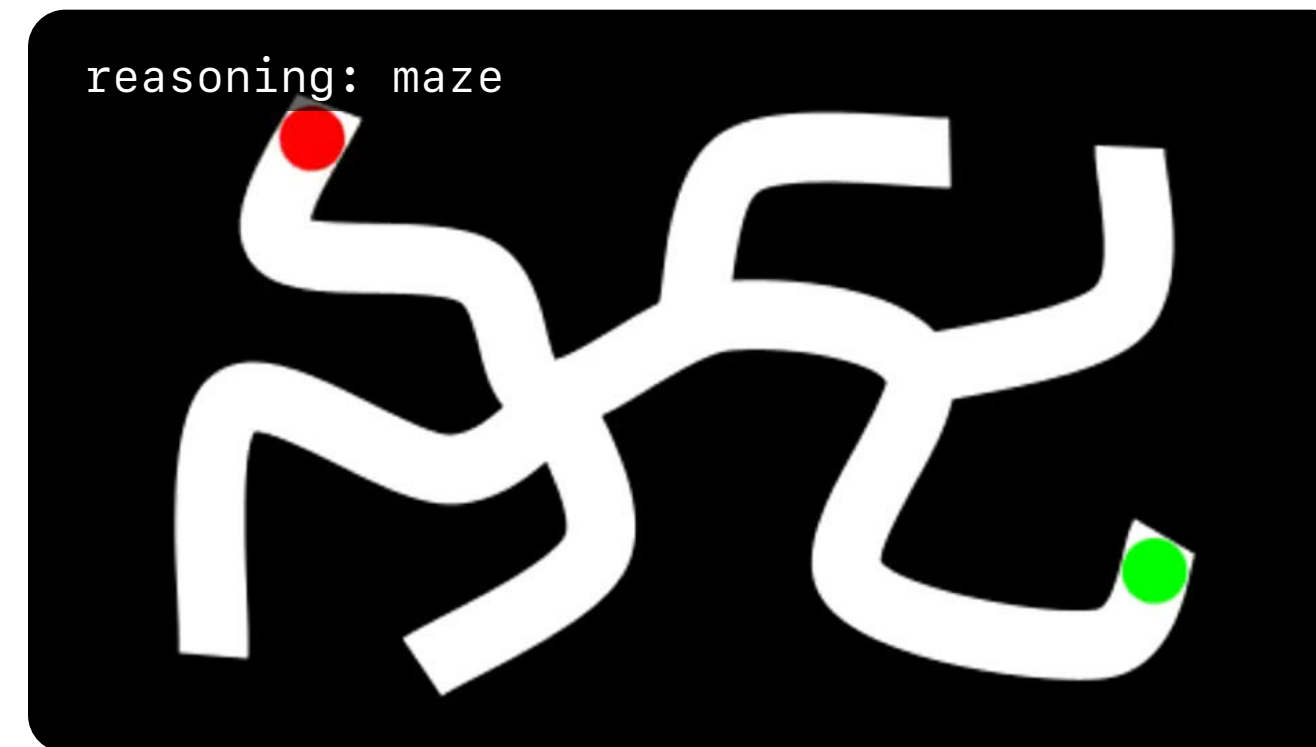
Video: Matrix-Game 2.0 project page



Video: HY-WorldPlay GitHub (Tencent Hunyuan)

→ Open real-time world models followed within months, and several reuse the Self Forcing recipe.

A video model can solve **vision tasks zero-shot**



Video: Wiedemer et al., 2025 · Veo 3 outputs

18,384

GENERATED VIDEOS ·
62 QUALITATIVE + 7 QUANTITATIVE TASKS

Chain-of-frames

frame-by-frame generation as step-by-step visual reasoning

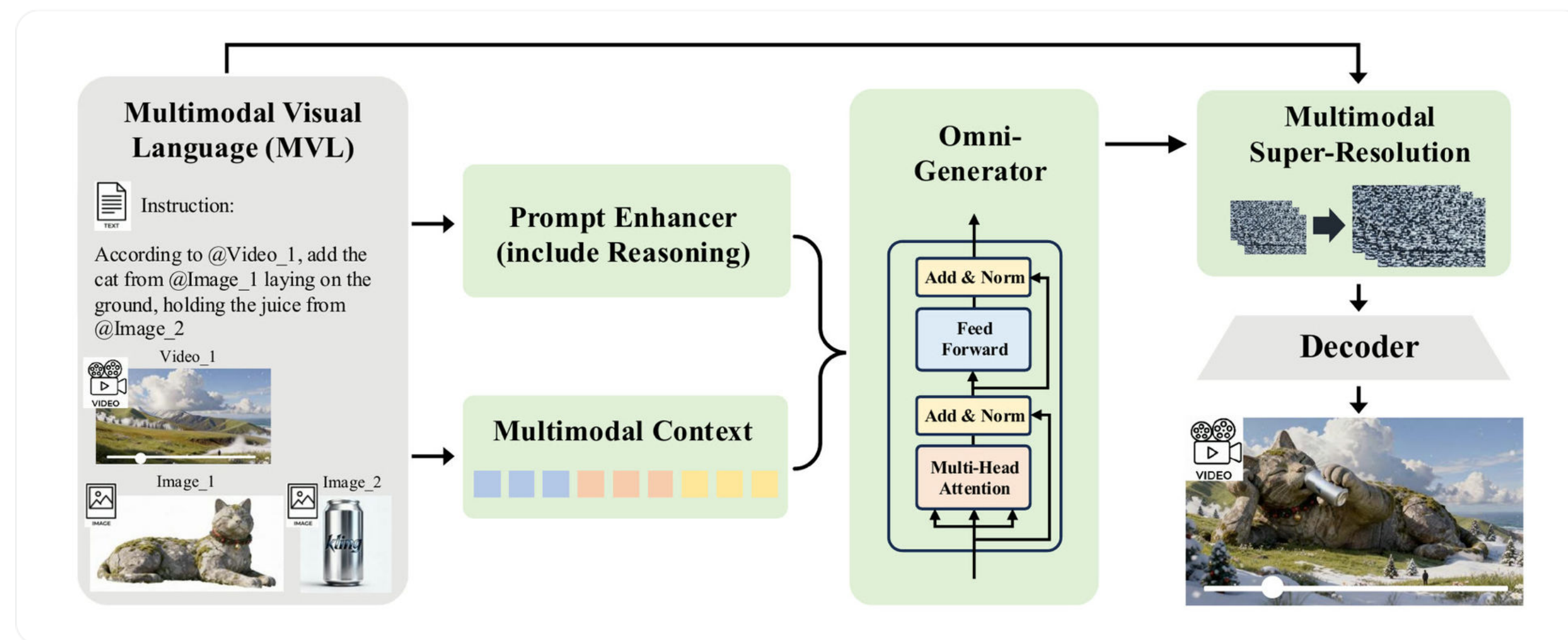
API has an LLM prompt rewriter; specialists still better

→ One image plus one instruction in, a video out: the answer is in the frames, as with LLM prompting.

From one prompt to **any input, many turns**



Video: Google, Gemini Omni (2026) · 3 of 4 turns, side by side (adapted)



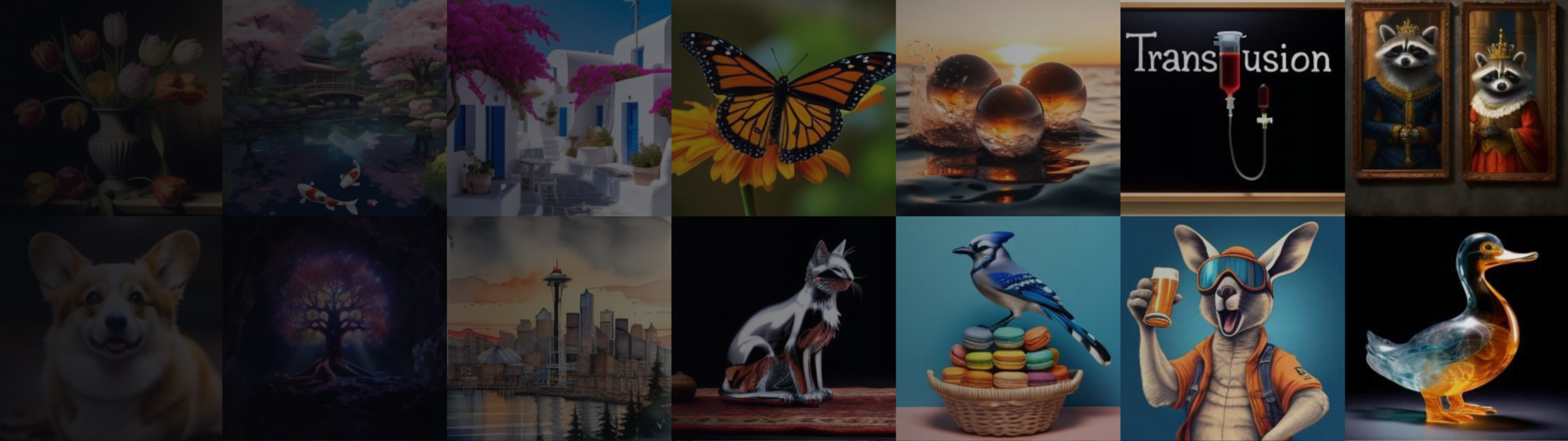
Kling-Omni: refs tagged in the prompt (@Image_1); coarse structure only

Seedance 2.0: text + image + video + audio in, 4–15 s out

backbones undisclosed · Seedance 2.0, Gemini Omni

Kling Team, 2025 · Fig. 1

→ The interface moved from one prompt to any references plus a multi-turn conversation.



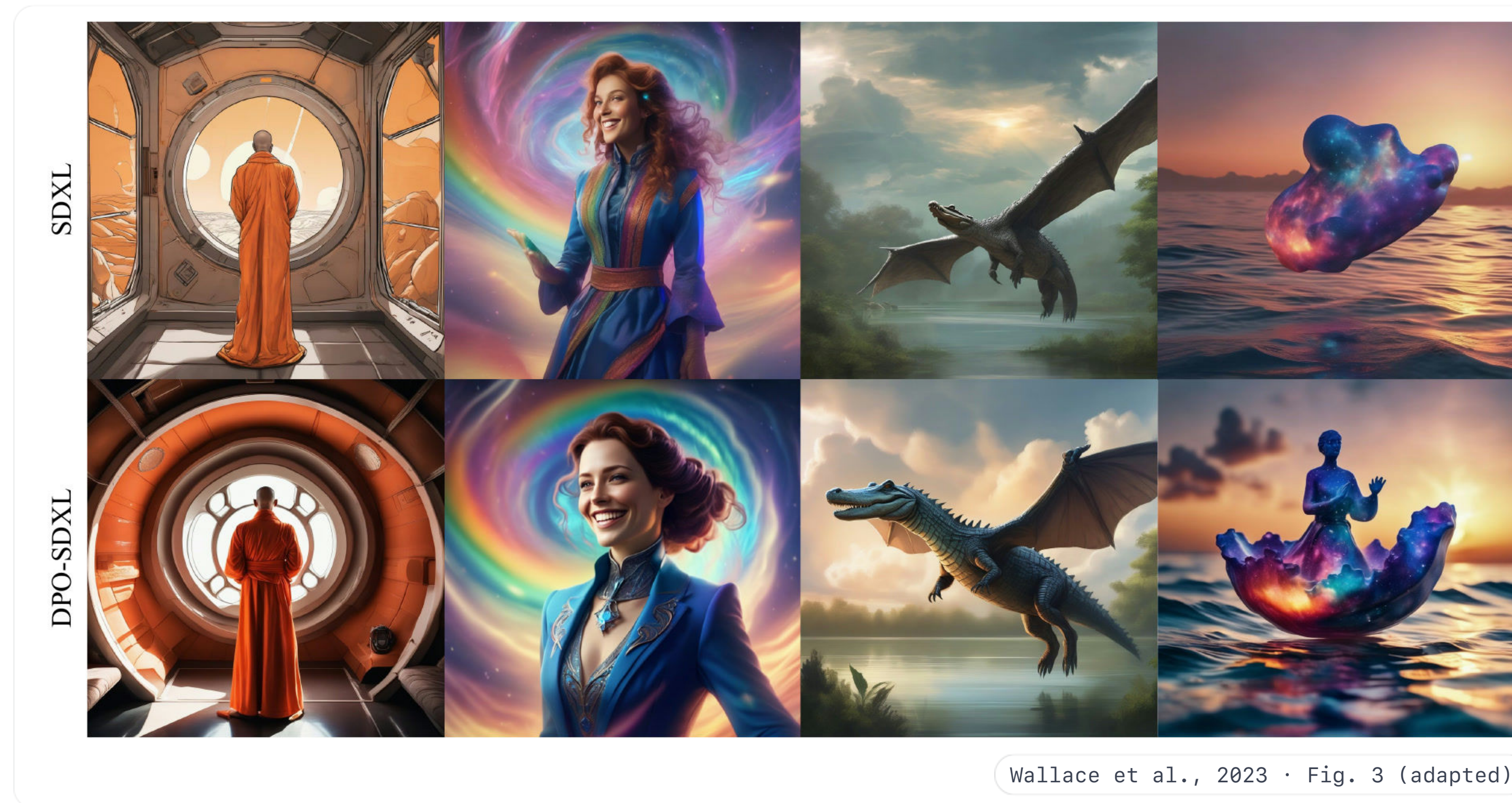
— CHAPTER 9 · 2023 → 2026

Diffusion meets the language model

- Diffusion-DPO
- Flow-GRPO
- MAR
- Transfusion
- GPT-4o image
- Nano Banana
- thinking image models
- beyond vision

Samples: Zhou et al., 2024 · Transfusion 7B

Align to human taste **without a reward model**



DIFFUSION-DPO LOSS (SIMPLIFIED)

$$\Delta(x) = \|\epsilon - \epsilon_{\theta}(x_t)\|^2 - \|\epsilon - \epsilon_{\text{ref}}(x_t)\|^2$$

$$\mathcal{L} = -\log \sigma(-\beta [\Delta(x^w) - \Delta(x^l)])$$

Δ : error vs. frozen reference, pushed lower for winner x^w than loser x^l ; $\mathbb{E}_t$, weighting omitted

851K

PICK-A-PIC PAIRS · SDXL

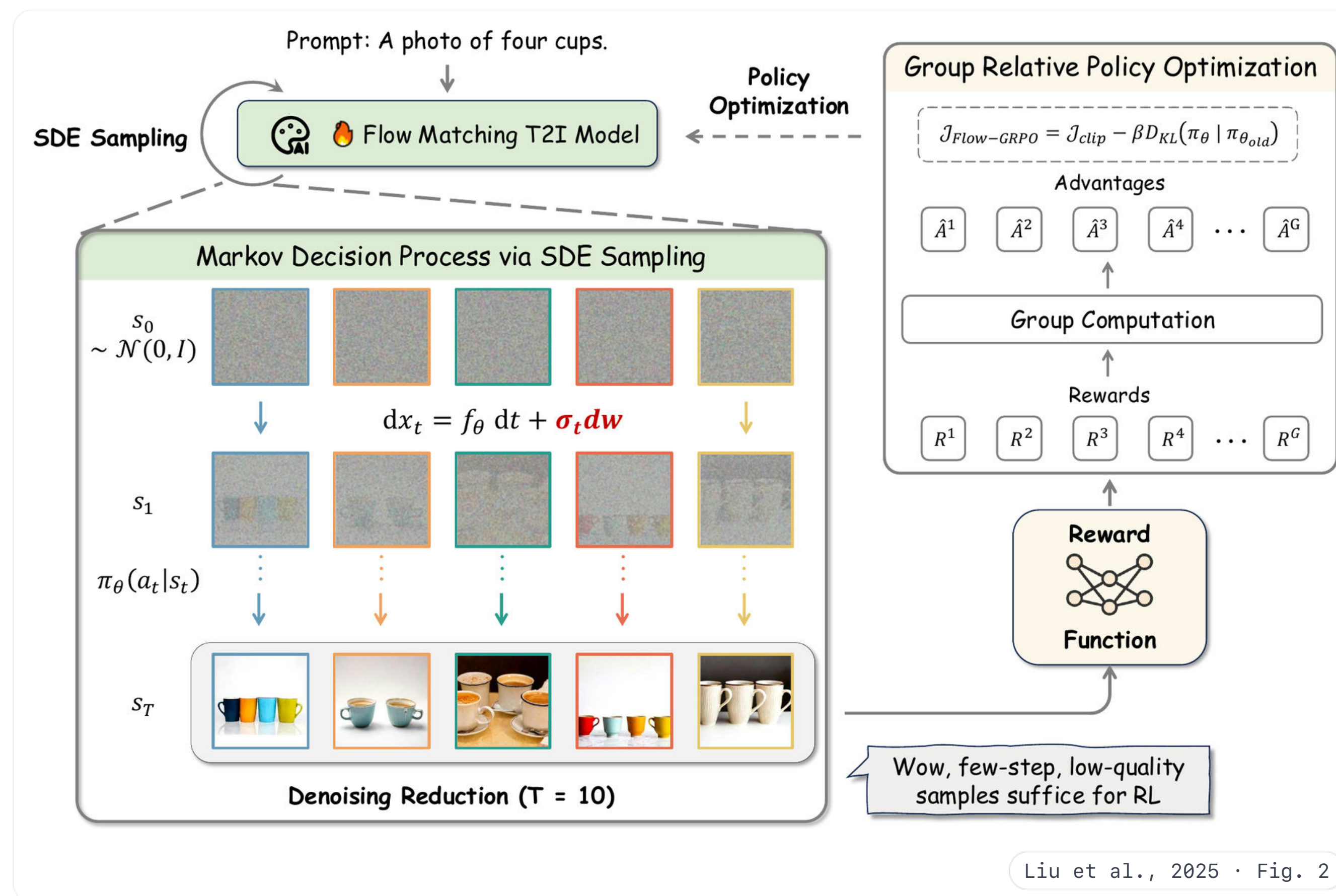
ReFL · backprop a differentiable reward

DDPO · online RL, any reward

Diffusion-DPO · offline pairs, no sampling

→ Show which of two images people preferred; the model learns to denoise the winner better.

An SDE turns flow sampling into a trainable policy



SAME MARGINALS AS THE ODE

$$dx_t = \left[v_t(x_t) - \frac{1}{2} g_t^2 \nabla \log p_t(x_t) \right] dt + g_t dw$$

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\})}{\text{std}(\{r_j\})}$$

SD3 convention, $t: 1 \rightarrow 0$. g_t (σ_t in the figure) sets the injected noise; $g_t = 0$ gives back the ODE. Group-relative advantage: no value network.

63→95%

GENEVAL

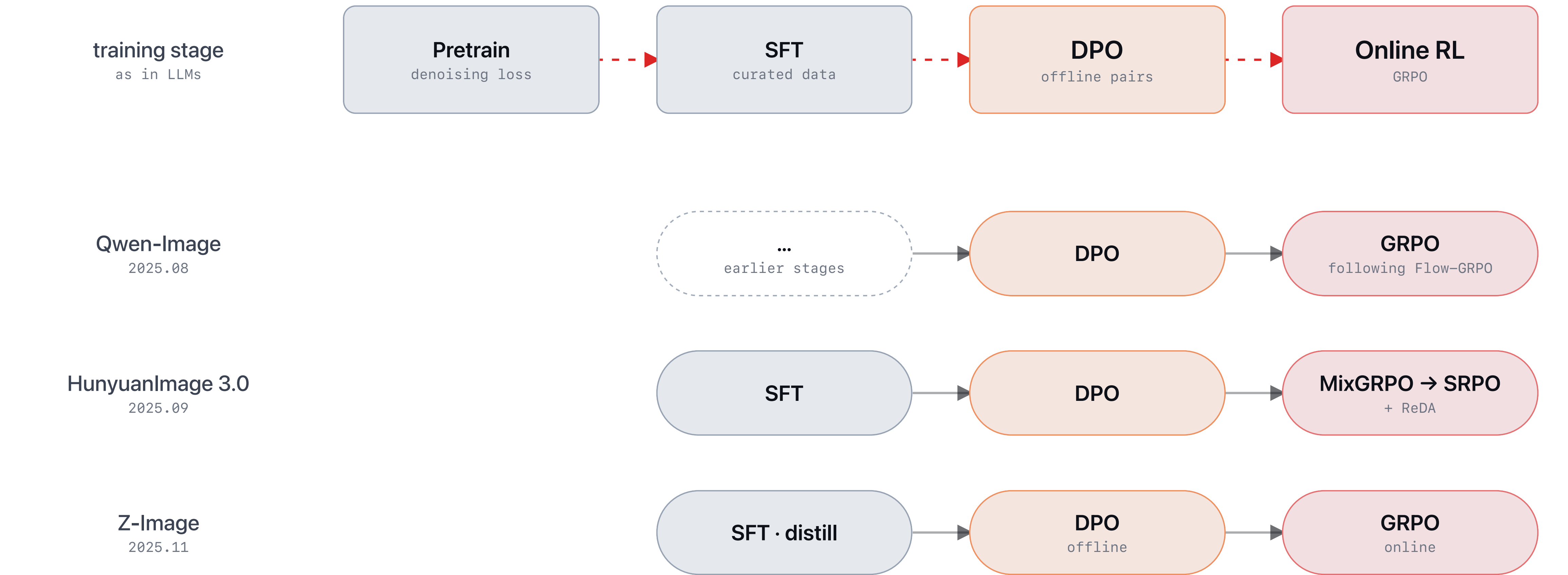
59→92%

TEXT RENDERING (OCR)

Liu et al., 2025 · Fig. 2

→ Sample several images per prompt, reinforce the above-average ones; deploy with the usual sampler.

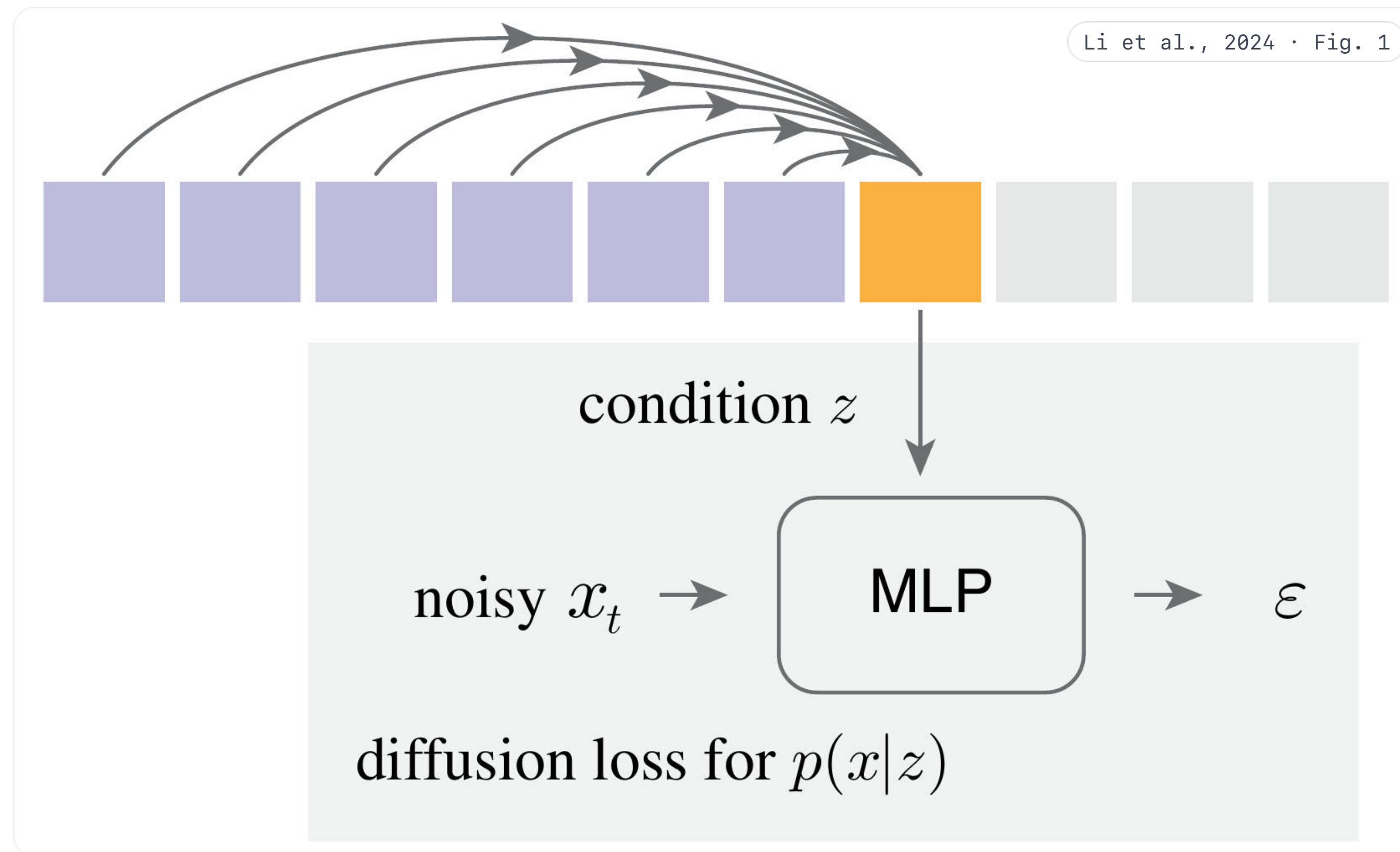
Image models now post-train like LLMs



Since: MixGRPO · TempFlow-GRPO · Pref-GRPO · DiffusionNFT

➔ Open image models now follow the LLM recipe: pretrain, fine-tune, learn preferences, then RL online.

Drop the codebook: a tiny diffusion head per token



DIFFUSION LOSS, PER TOKEN

$$\mathcal{L}(z, x) = \mathbb{E}_{\epsilon, t} \left\| \epsilon - \epsilon_{\theta}(x_t \mid t, z) \right\|^2$$

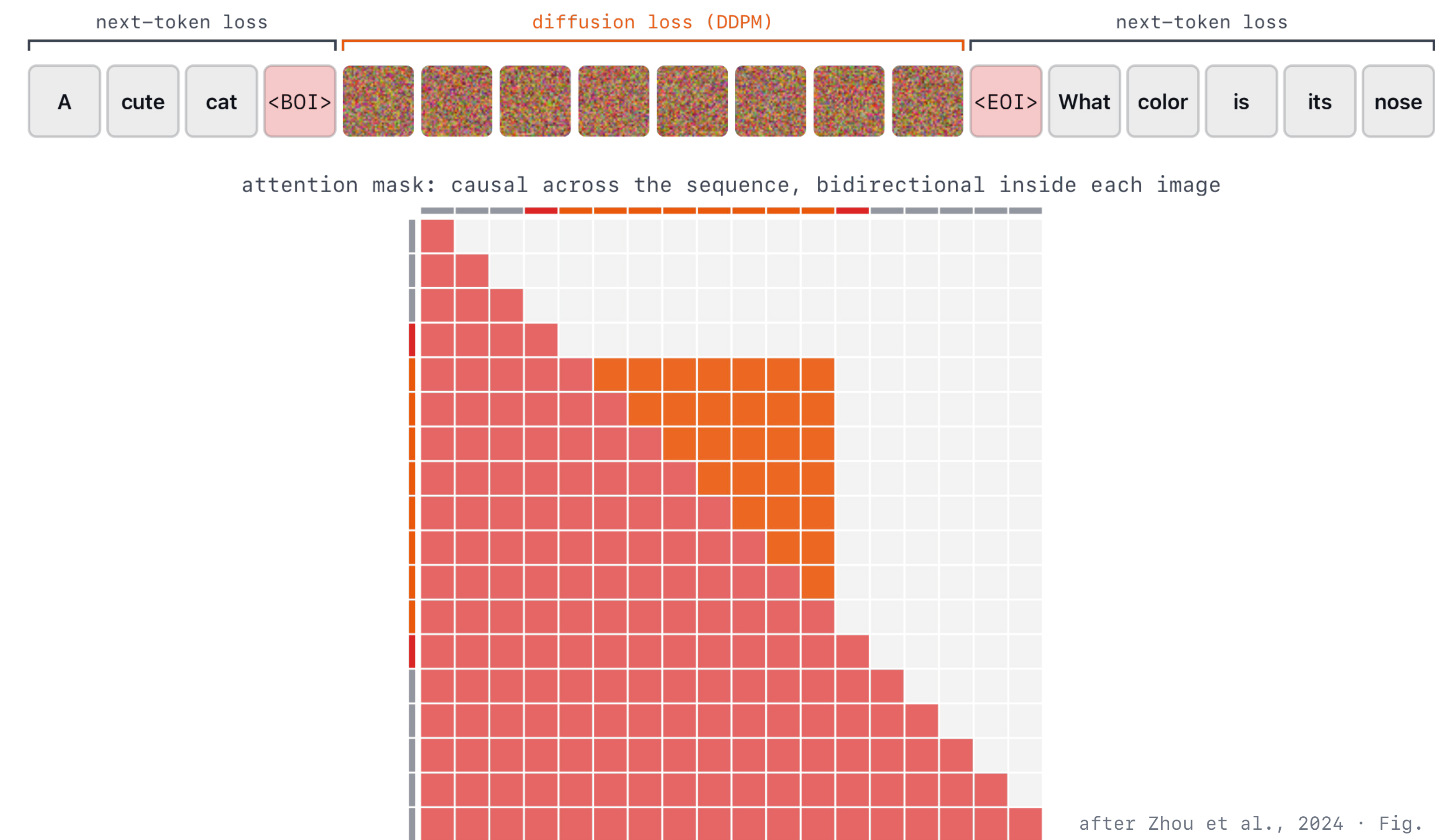
The transformer gives z ; a small MLP denoises the continuous token x . It replaces the softmax over a VQ codebook.



Samples: Li et al., 2024 · MAR-H, ImageNet 256²

→ The transformer decides what comes next; a small diffusion head draws its exact continuous value.

One model predicts text and denoises image patches



ONE LOSS, ONE SET OF WEIGHTS

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \lambda \mathcal{L}_{\text{DDPM}}$$

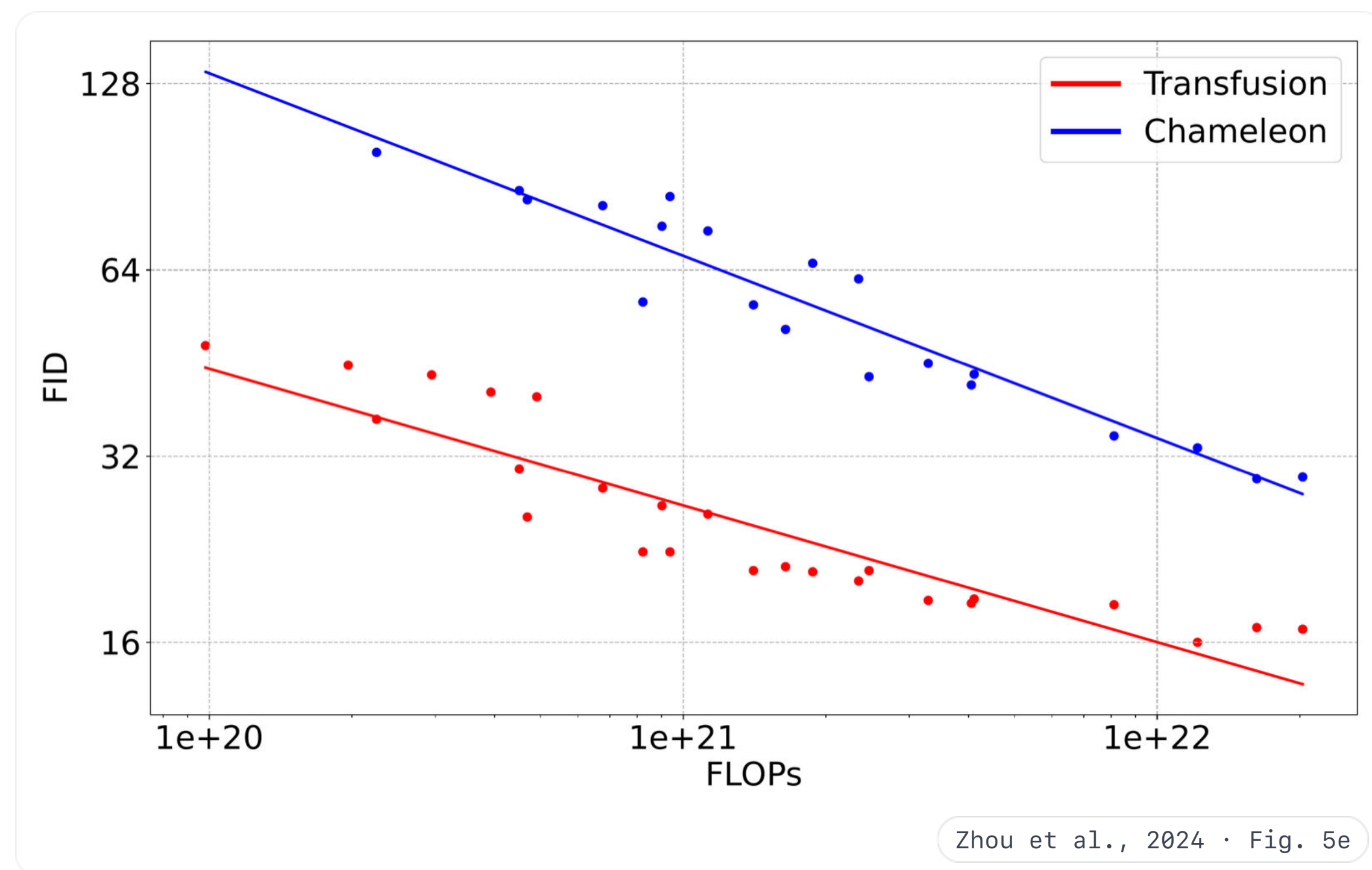
cross-entropy on text + noise MSE on VAE-latent patches, $\lambda = 5$

AT INFERENCE

sample text → <BOI> (begin image) →
append n noise patches → denoise T steps
→ <EOI> (end) → back to text

→ No separate image model: the same weights read the conversation and denoise the picture inside it.

In controlled tests, diffusion beat **discrete image tokens**



Lower FID at every scale · same params, data, compute



Samples: Zhou et al., 2024 · Transfusion 7B

7B

FROM SCRATCH

2T

TOKENS

<1/3

CHAMELEON'S COMPUTE

Before it, MAR (2024.06): per-token diffusion head, no VQ · later: JanusFlow · BAGEL · HunyuanImage 3.0

→ Continuous patches with a diffusion loss beat discrete image tokens, using under a third of the compute.

Image generation moves **inside the chat model**



GPT-4o outputs

Chen et al., 2025 · Fig. 5 (GPT-4o outputs, adapted)

DISCLOSED

joint image-text training · aggressive
post-training · 10–20 objects

not disclosed

image tokens? diffusion decoder? (the “→
diffusion → pixels” whiteboard is a sample)

2.1 Safety Challenges: New risks from native image generation

Unlike DALL·E, which operates as a diffusion model, **4o image generation is an autoregressive model natively embedded within ChatGPT**. This fundamental difference introduces several new capabilities that are distinct from previous generative models, and that pose new risks:

OpenAI, 2025 · GPT-4o system card addendum, p. 1 (highlight added)

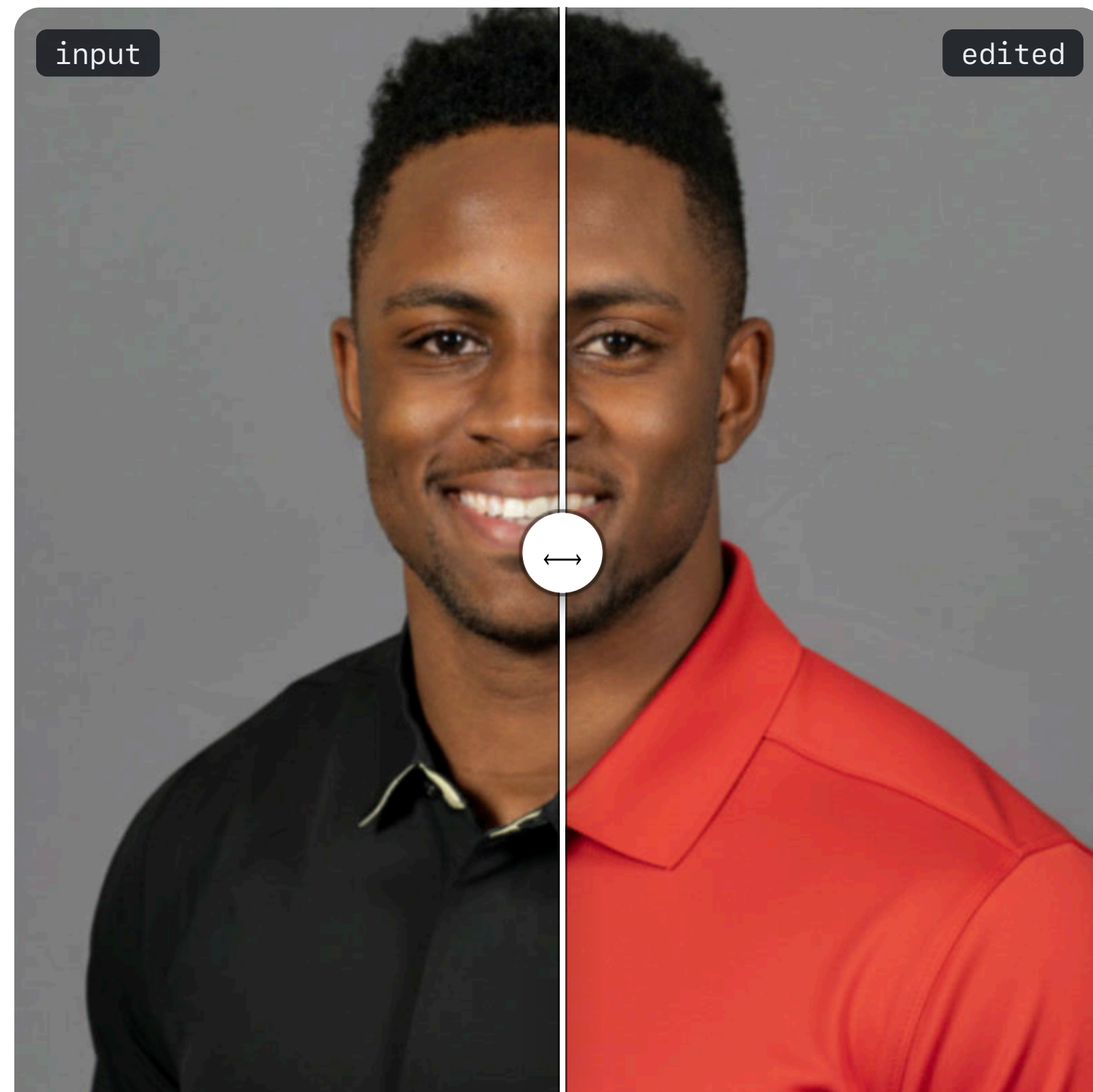
>700M

IMAGES BY >130M USERS, WEEK ONE (PER OPENAI)

Open frontier still diffusion/flow: Qwen-Image · FLUX.2 · Z-Image

→ Architecture per OpenAI: “an autoregressive model”; how its pixels are rendered is not public.

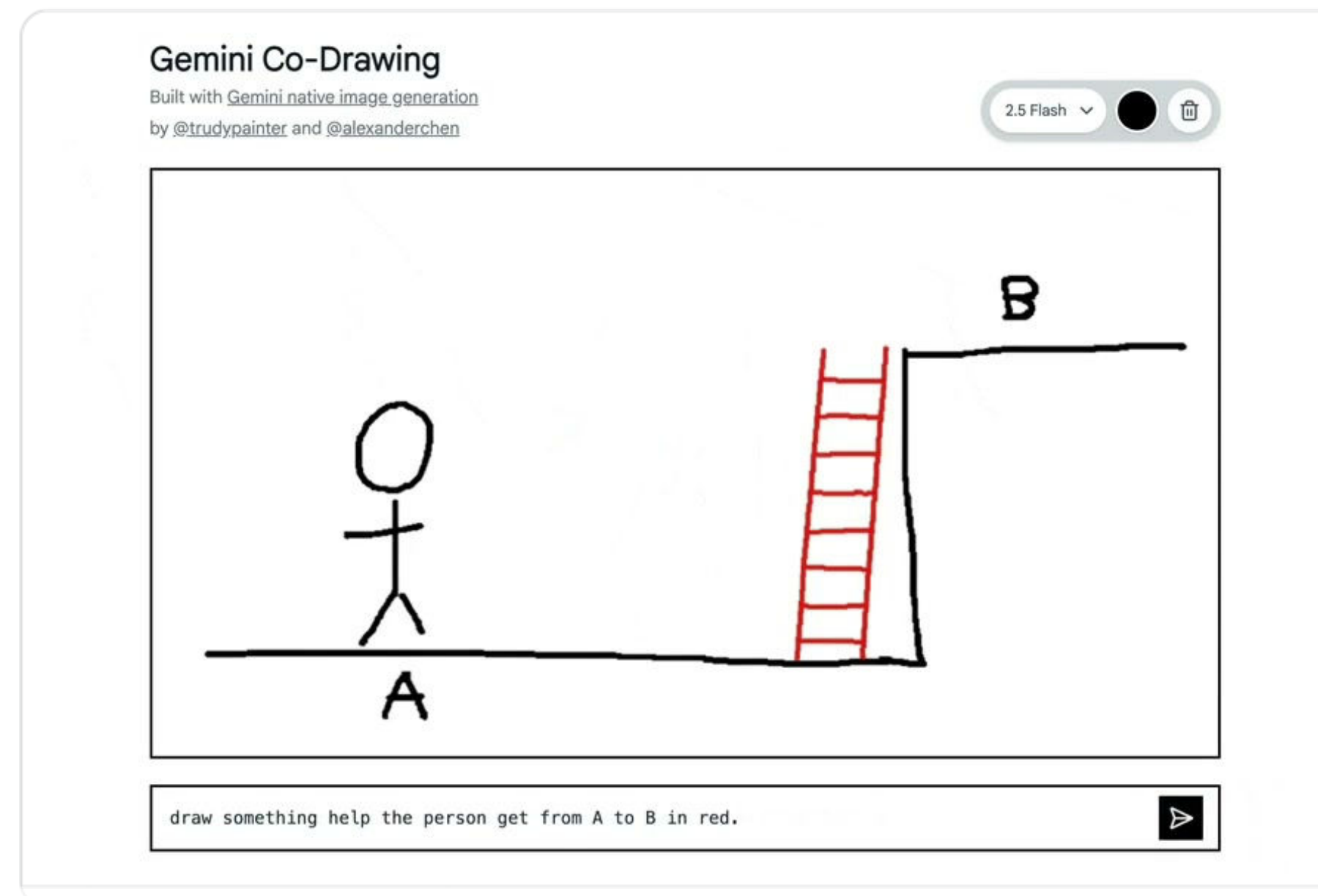
Edit by conversation, **keep everything else**



Google, Nano Banana (2025) · Pixshop example, split (adapted)

"change my shirt color to red and remove earring"

→ Say what to change; face, pose and background stay as they were.



Video: Google, Nano Banana (2025)

World knowledge: sketch + "solve for x" → answer in red

Nov 2025: Pro adds reasoning + Search
Where disclosed, pixels are still mostly diffusion/flow

\$0.039

PER IMAGE

>200M

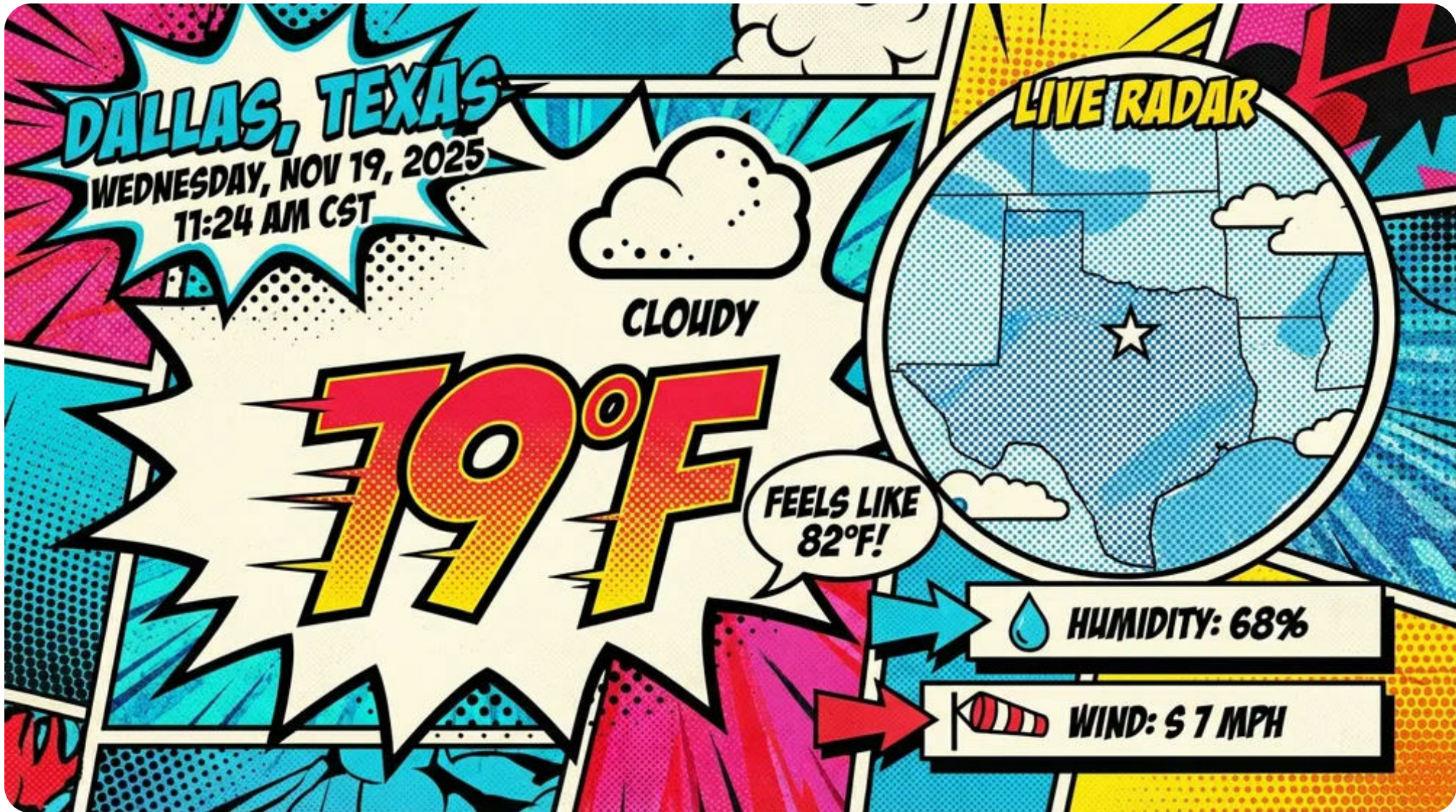
EDITS (GOOGLE-REPORTED)

architecture undisclosed

Image models that **reason, search, then draw**



Google, Nano Banana Pro (2025)



live weather via Search

Google, Nano Banana Pro (2025)



translate EN → KO

Google, Nano Banana Pro (2025)

2025-26 PATTERN (NB PRO · GPT-IMAGE-2 · MUSE IMAGE)

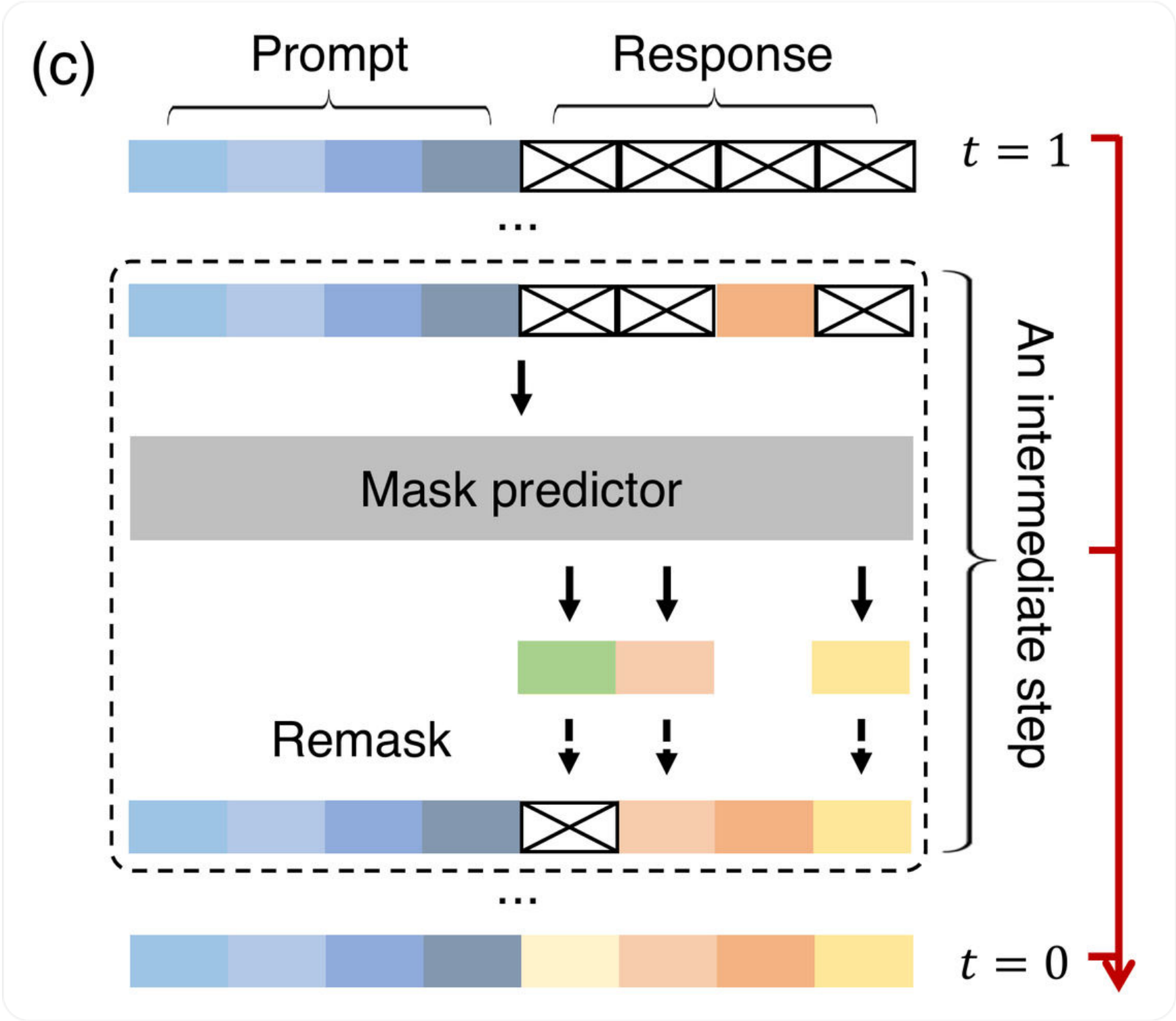
prompt → **reason** → **search** → draw → **check** → redraw

NB Pro: ≤14 input images · ≤5 people consistent · 2K/4K

architecture undisclosed

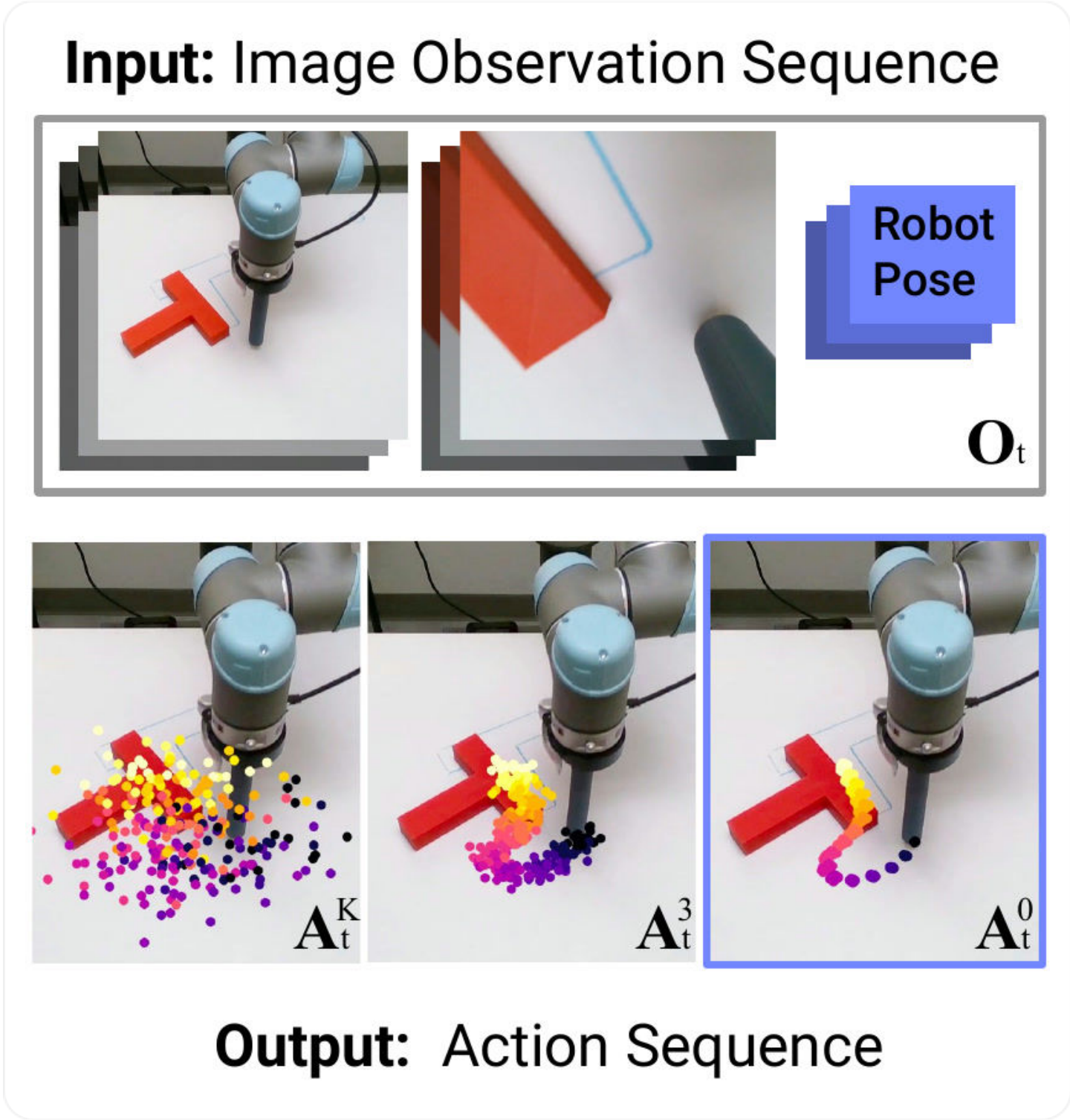
➔ The goal moved from pretty to correct and readable: plan, look things up, check, then redraw.

The same loop now writes **text** and **robot actions**



Nie et al., 2025 · Fig. 2c (adapted)

LLaDA 8B: unmask from all-[MASK]



Chi et al., 2023 · Fig. 2a (adapted)

Diffusion Policy: denoise an action chunk

1109	TOK/S · MERCURY CODER MINI, H100
1479	TOK/S · GEMINI DIFFUSION
2146	TOK/S · SEED DIFFUSION, H20
as reported, not like-for-like	

→ For text the “noise” is [MASK]; for robots it is a noisy action sequence. The loop is the same.

— TRENDS · 2015 → 2026

Ten trends, three takeaways

then → now

video replays image

AR across · diffusion within

three takeaways

open questions

→ Nine chapters of milestones, joined into a few lines you can remember.

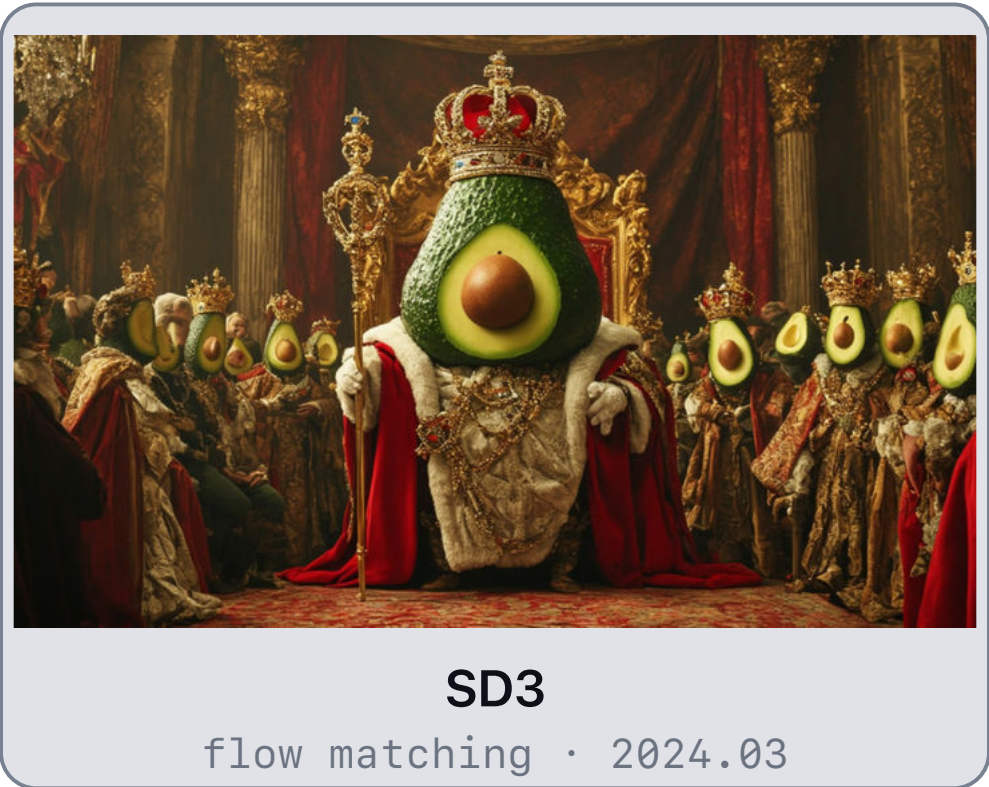
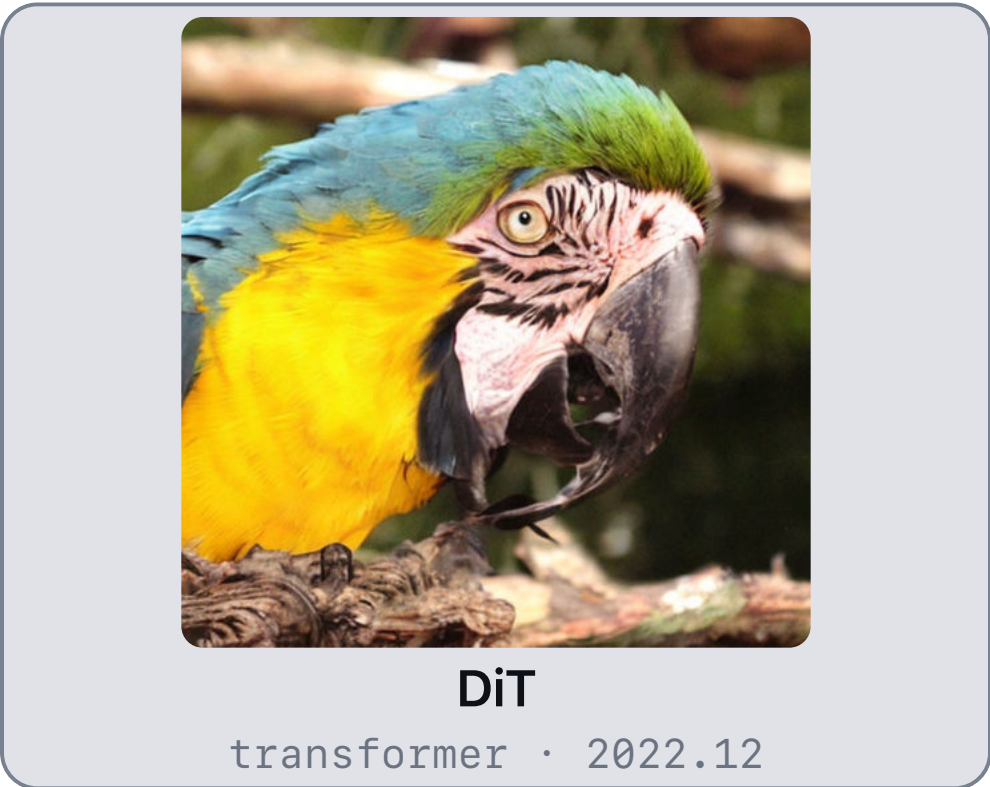
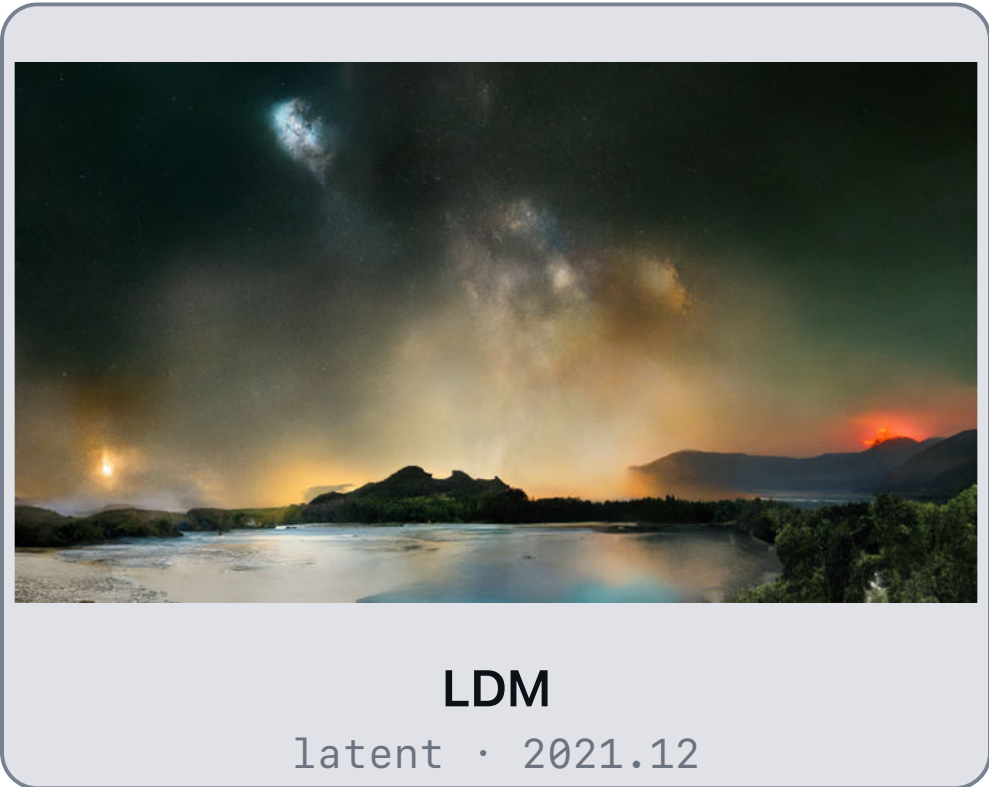
Montage: DDPM 2020 · LDM 2021 · Sora 2024 · Self Forcing 2025 · Genie 3 2025 (arch. undisclosed)

Every design choice moved: then → now

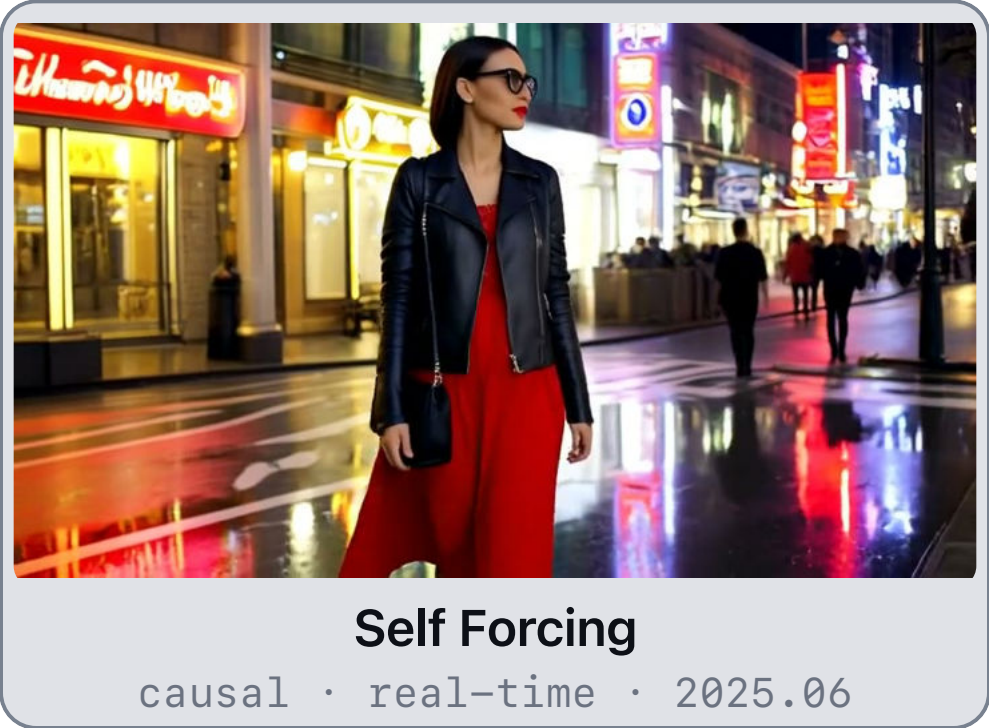
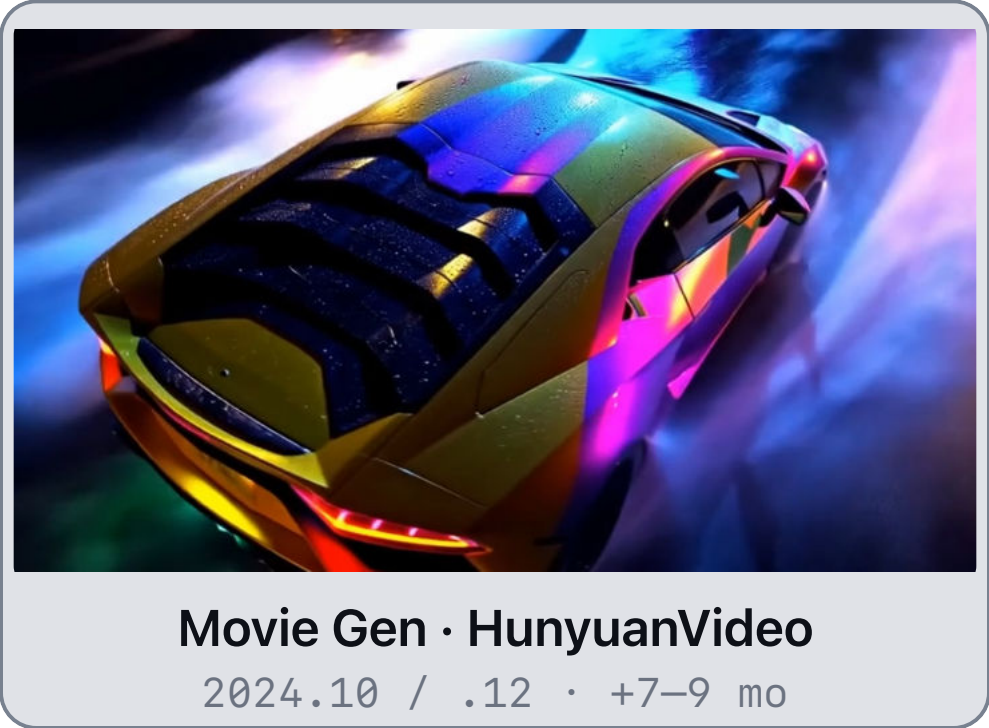
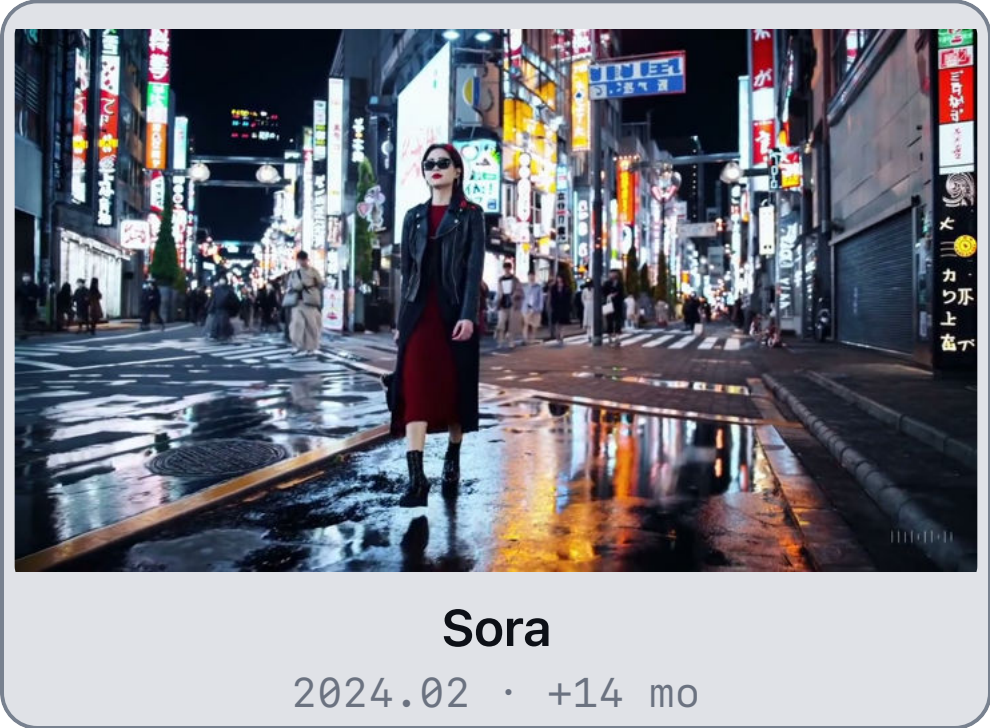
Math ①	DDPM · score → SDE / ODE → EDM → flow matching	ch 1 · 2 · 5
Backbone	U-Net → DiT → MMDiT	ch 5 · 7
Denoise in	pixels + cascade → VAE latent → semantic latent or pixels?	ch 3 · 5
Text encoder	class label → CLIP → T5 → (multimodal) LLM	ch 3 · 5 · 9
Data ②	alt-text pairs → dense recaptions	ch 3 · 7
Steps	~1000 → ~50 → 1-4 → real-time video	ch 1 · 6 · 8
Control	sampling-time guidance (CFG) → adapters → everything in context	ch 2 · 4
Modality	image → video → interactive worlds	ch 7 · 8
Post-training	SFT → DPO → GRPO	ch 9
vs. autoregression ③	rivals → division of labour	ch 8 · 9

→ Almost every design choice changed; the core idea, learn to undo noise, did not.

Video replays the image playbook, 7–16 months late



...then its own path
Diffusion Forcing → CausVid → Self Forcing



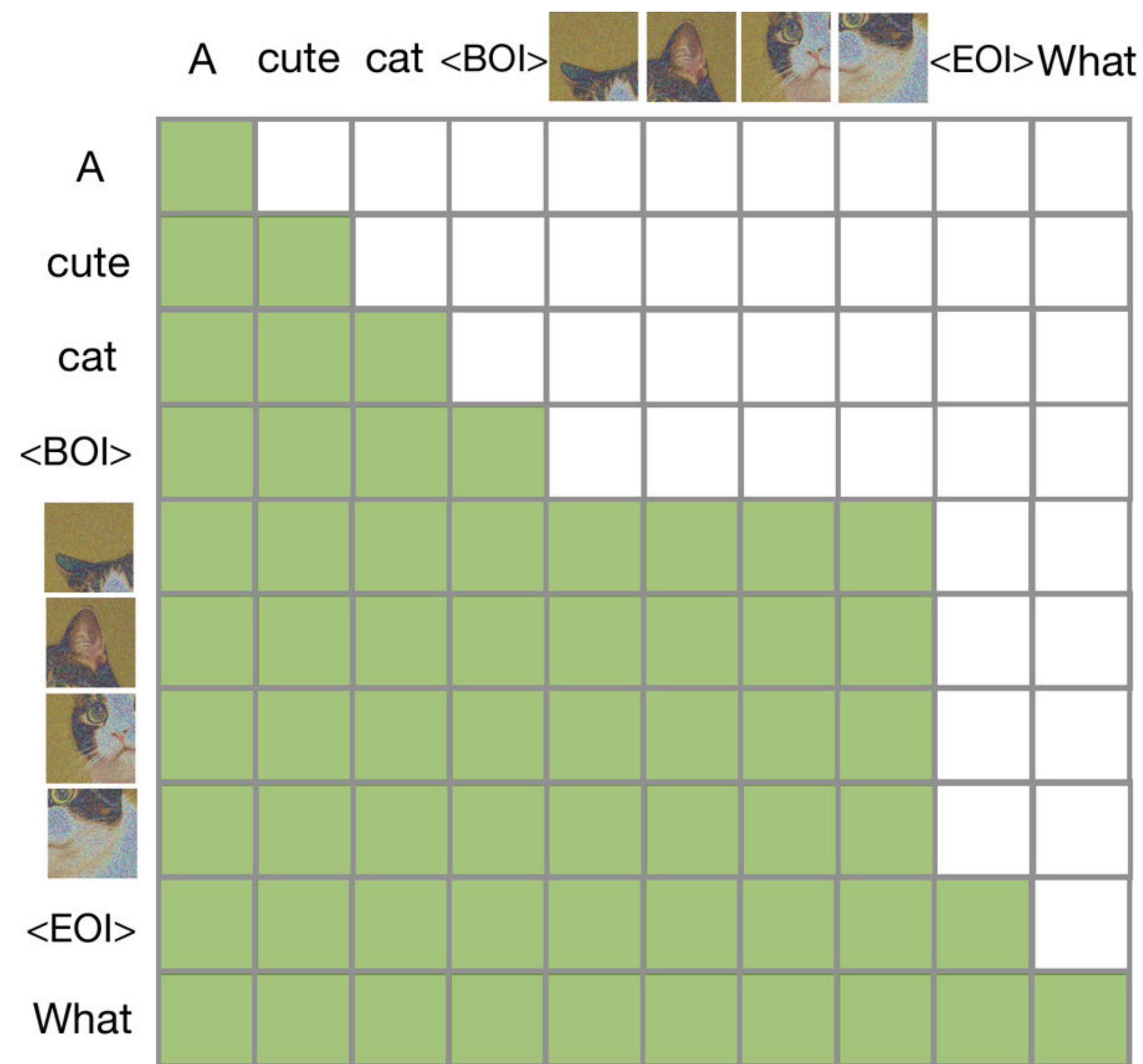
Samples (cropped): LDM · DiT · SD3 · Video LDM papers · Sora: OpenAI · HunyuanVideo GitHub · Self Forcing page

Top: image paper · bottom: its video counterpart · Sora objective undisclosed

→ Video borrowed latents, transformers and flow matching from images, then went causal and real-time.

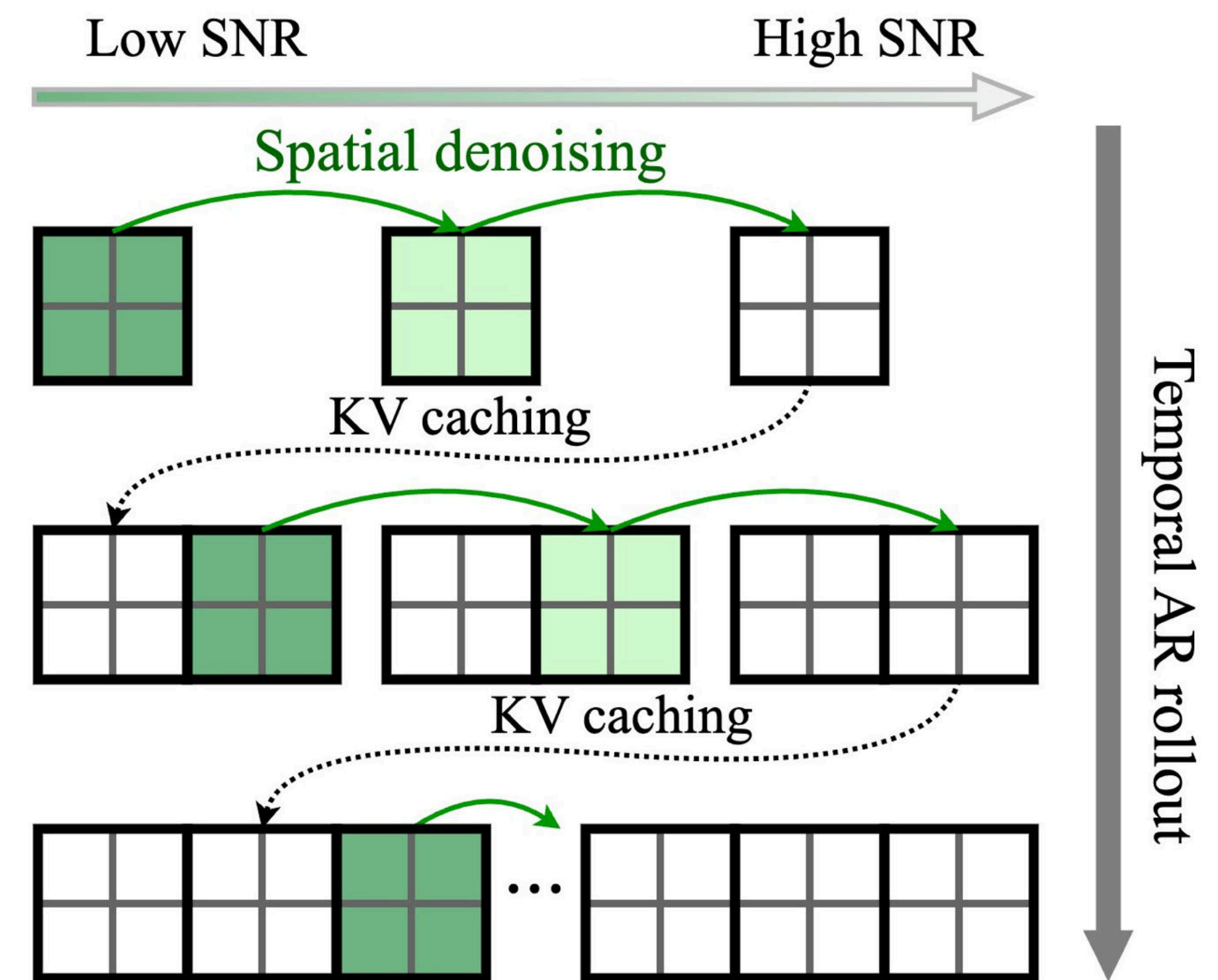
AR goes **across**, diffusion goes **within**

speaker's interpretation, not a claim from any paper



Zhou et al., 2024 · Fig. 4

Transfusion: causal across tokens, bidirectional within an image



Self Forcing project page · teaser, right half (adapted)

Self Forcing: AR rollout across frames, denoising within each

→ One reading: AR orders the pieces; diffusion fills in the continuous values inside each piece.

Three trends to take home

1



Gao et al., 2024 · blog header (crop)

One family; flow matching by default
diffusion = score = flow · SD3, FLUX

2

alt-text: "now at victorian plumbing.co.uk"



Betker et al., 2023 · Fig. 3 (adapted)

Data & conditioning drove the leaps
recaptioning · T5 → LLM encoders

3

streamed in real time



Video: Self Forcing project page

Diffusion as the renderer
one reading · inside LLMs · real-time video & worlds

→ One model family; data and conditioning drove the big leaps; diffusion increasingly acts as the renderer.

Four questions still open

Li & He, 2025 · Fig. 1

Which latent?
VAE · semantic (RAE) · pixels
impact undetermined

Geng et al., 2025 · Fig. 1 (adapted)

One step from scratch?
MeanFlow: FID 3.43 at 1 step

Video: Google DeepMind, Genie 3 (2025)

Memory beyond minutes?
avoiding drift is open
Genie 3: architecture undisclosed

N=10 N=100000

Kadkhodaie et al., 2023 · Fig. 2 (crop)

Memorize or generalize?
disjoint 10^5 -image sets → same face

Also open: evaluation · cost · copyright

→ The recipe has converged; its latent, step count, long-range memory and theory have not.

Thank you

Questions?

→ From noise to worlds: one idea, learn to undo noise, scaled from 2015 to 2026.

t = 0.00 step 50/50

Background: LDM sample · Rombach et al., 2021 (denoising is illustrative)