NVIDIA

Tackling the Generative Learning Trilemma with Accelerated Diffusion Models

Arash Vahdat

DENOISING DIFFUSION MODELS (DDMS)

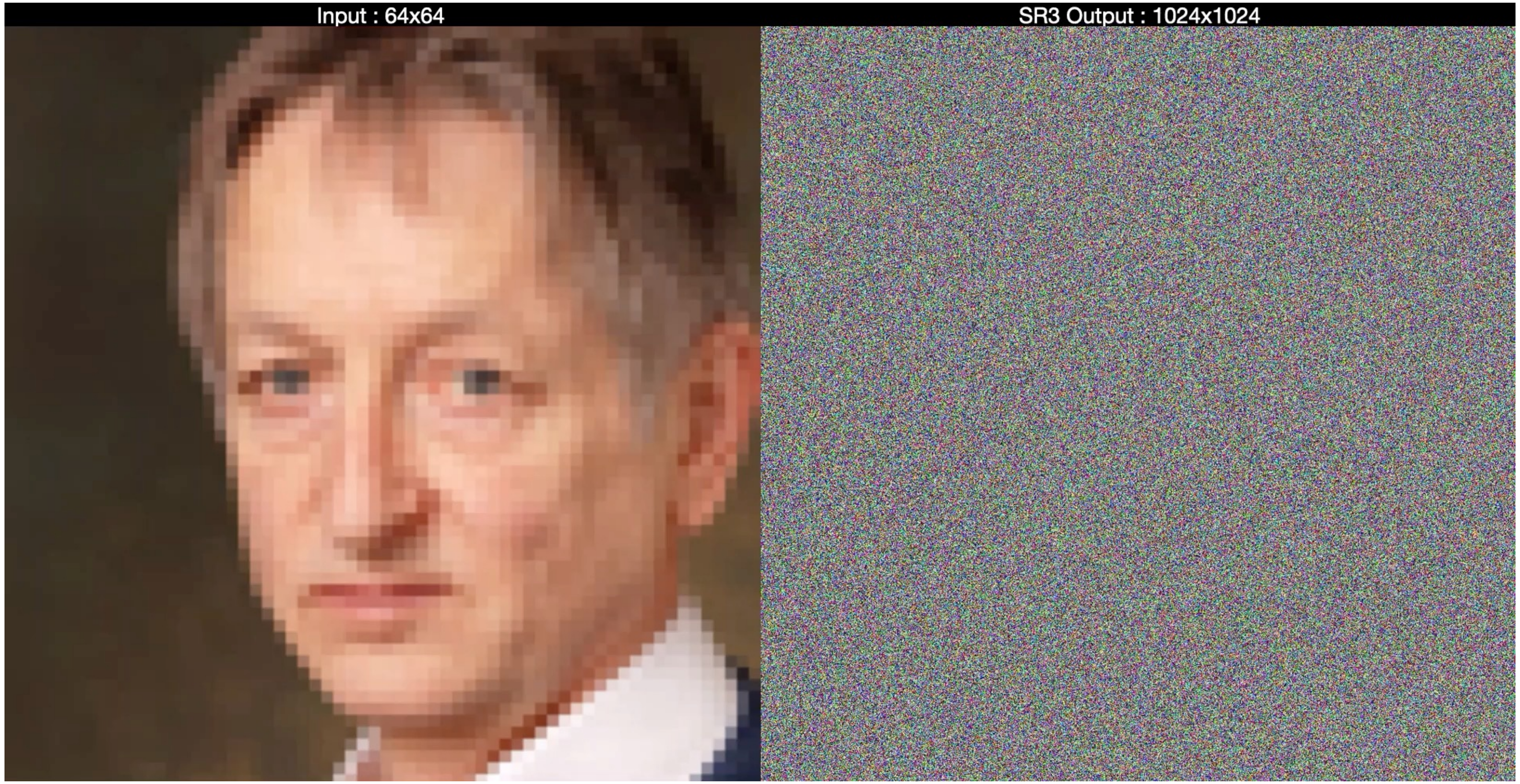
Emerging as powerful generative models, outperforming GANs

"Diffusion Models Beat GANs on Image Synthesis"
Dhariwal & Nichol, OpenAI, 2021

"Cascaded Diffusion Models for High Fidelity Image Generation"
Ho et al., Google, 2021

IMAGE SUPER-RESOLUTION

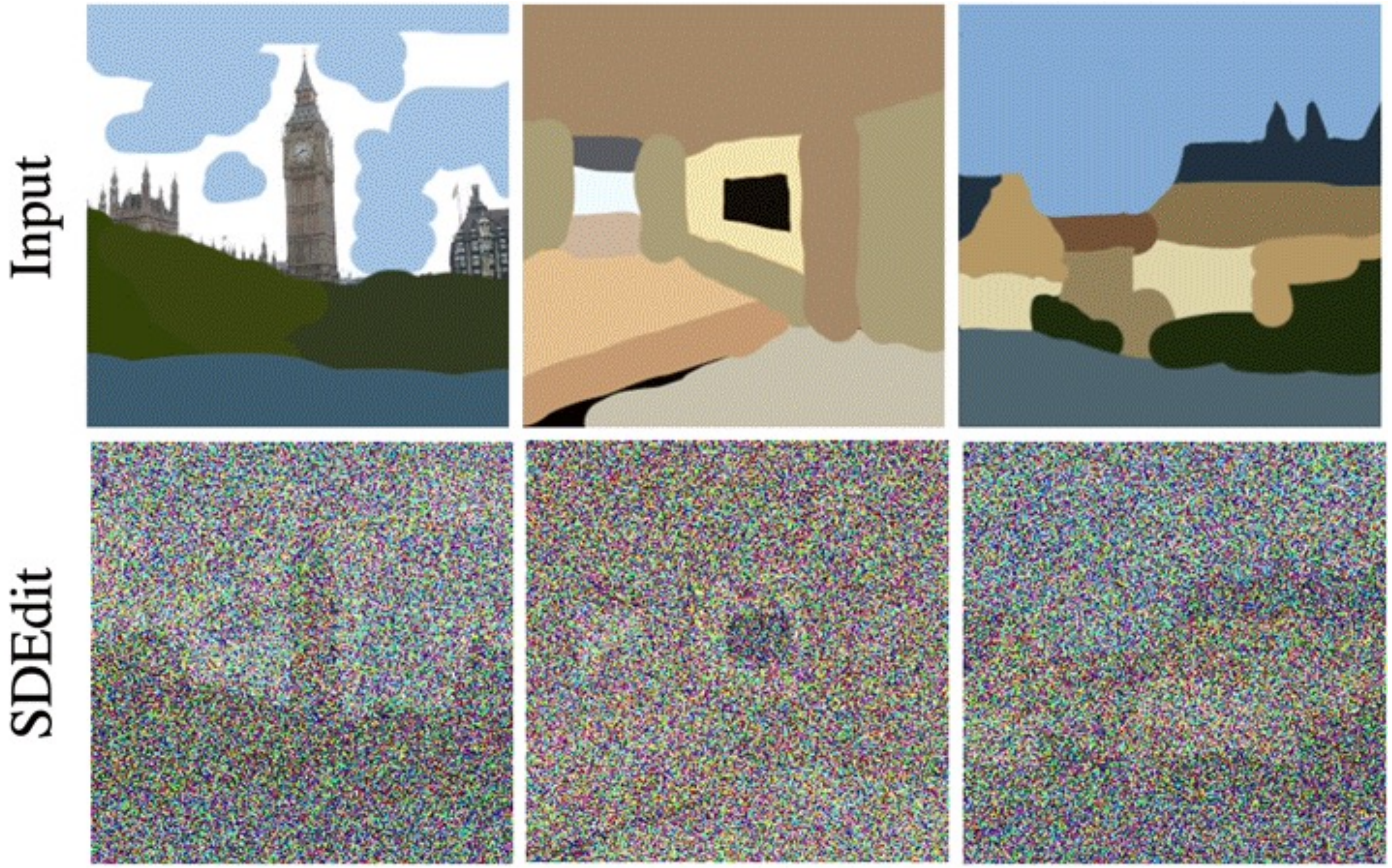
Successful applications



Saharia et al., Image Super-Resolution via Iterative Refinement, ICCV 2021

IMAGE EDITING

Successful applications



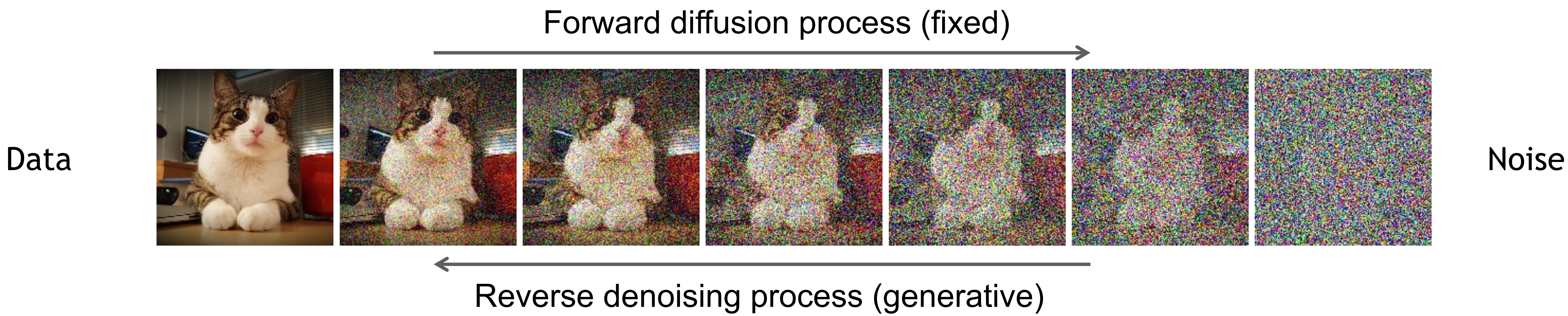
Meng et al., SDEdit: Image Synthesis and Editing with Stochastic Differential Equations, 2021

DENOISING DIFFUSION MODELS

Learning to generate by denoising

Denoising diffusion models (a.k.a score-based generative models) consist of two processes:

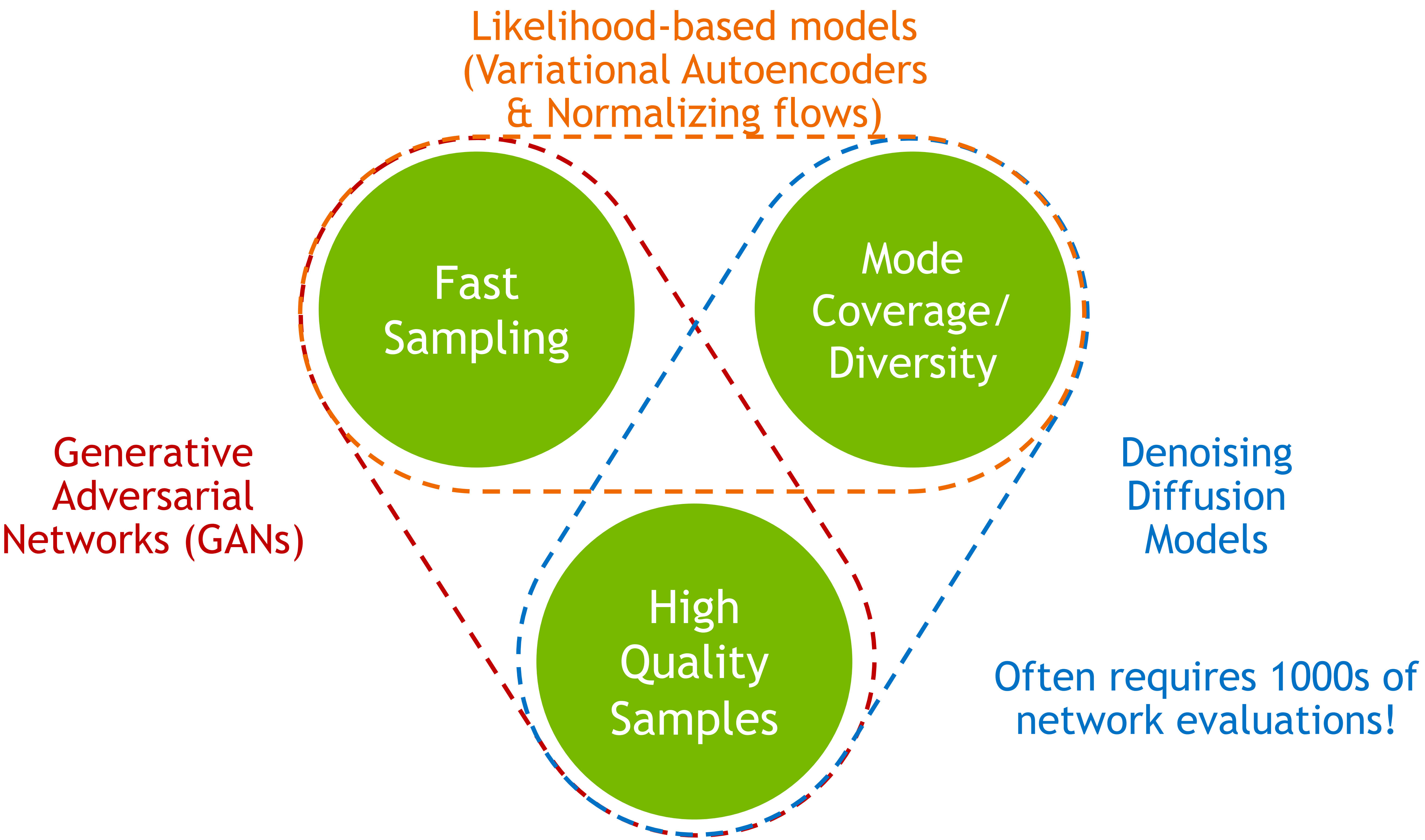
- Forward diffusion process that gradually adds noise to input
- Reverse denoising process that learns to generate data by denoising



Sohl-Dickstein et al., Deep Unsupervised Learning using Nonequilibrium Thermodynamics, ICML 2015
Ho et al., Denoising Diffusion Probabilistic Models, NeurIPS 2020
Song et al., Score-Based Generative Modeling through Stochastic Differential Equations, ICLR 2021

WHAT MAKES A GOOD GENERATIVE MODEL?

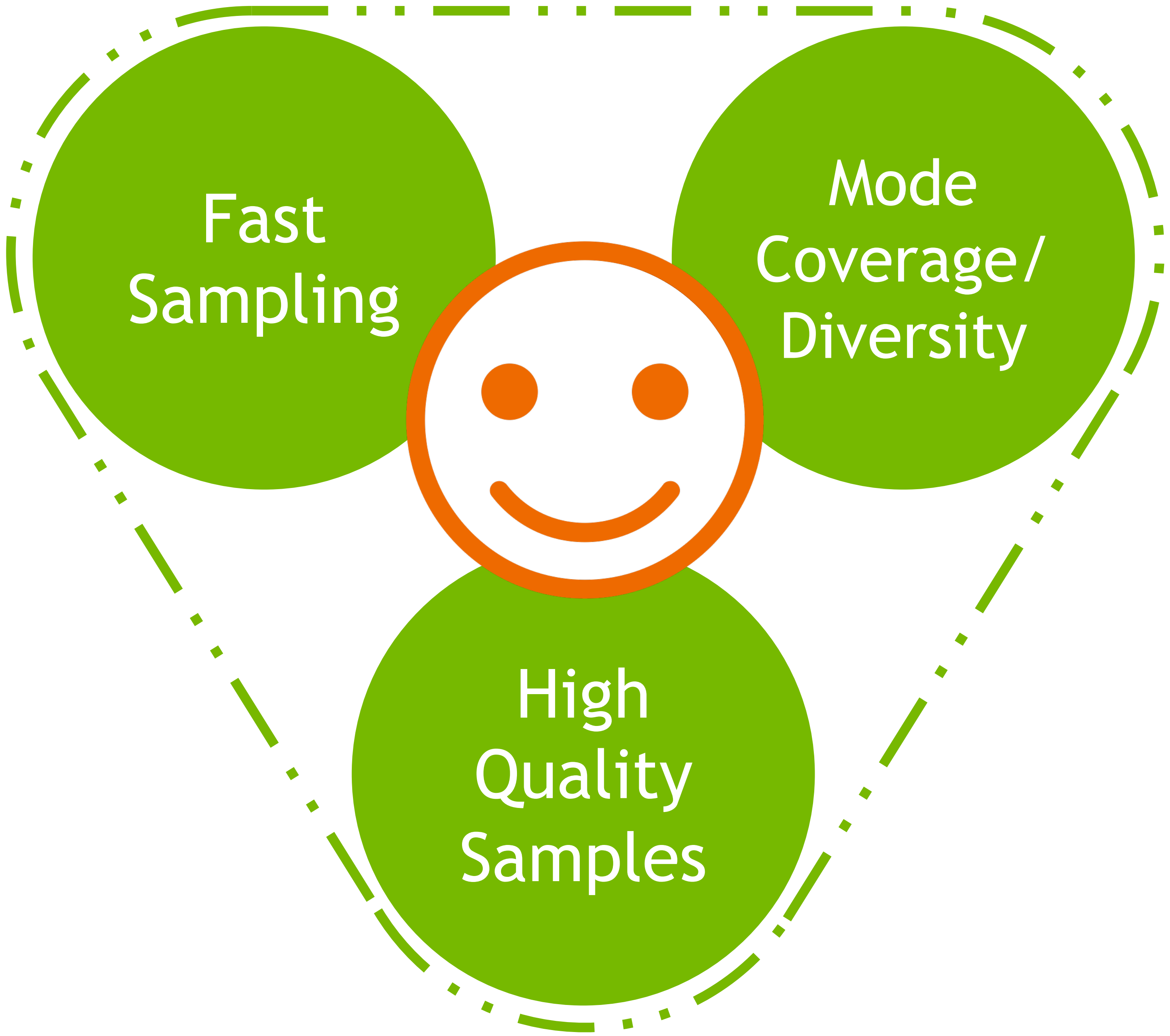
The generative learning trilemma



WHAT MAKES A GOOD GENERATIVE MODEL?

The generative learning trilemma

Tackle the trilemma by accelerating diffusion models



Score-based Generative Modeling in Latent Space

Arash Vahdat*, Karsten Kreis*, Jan Kautz

(*equal contribution)

NeurIPS 2021

DISCRETE-TIME DIFFUSION MODELS

Formal definition of forward and reverse processes in T steps:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t\geq 1} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \quad q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

Forward diffusion process (fixed)

Data

Noise

Reverse denoising process (generative)

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t\geq 1} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \quad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \underbrace{\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)}_{\text{Trainable network (U-net, Denoising Autoencoder)}}, \sigma_t^2\mathbf{I})$$

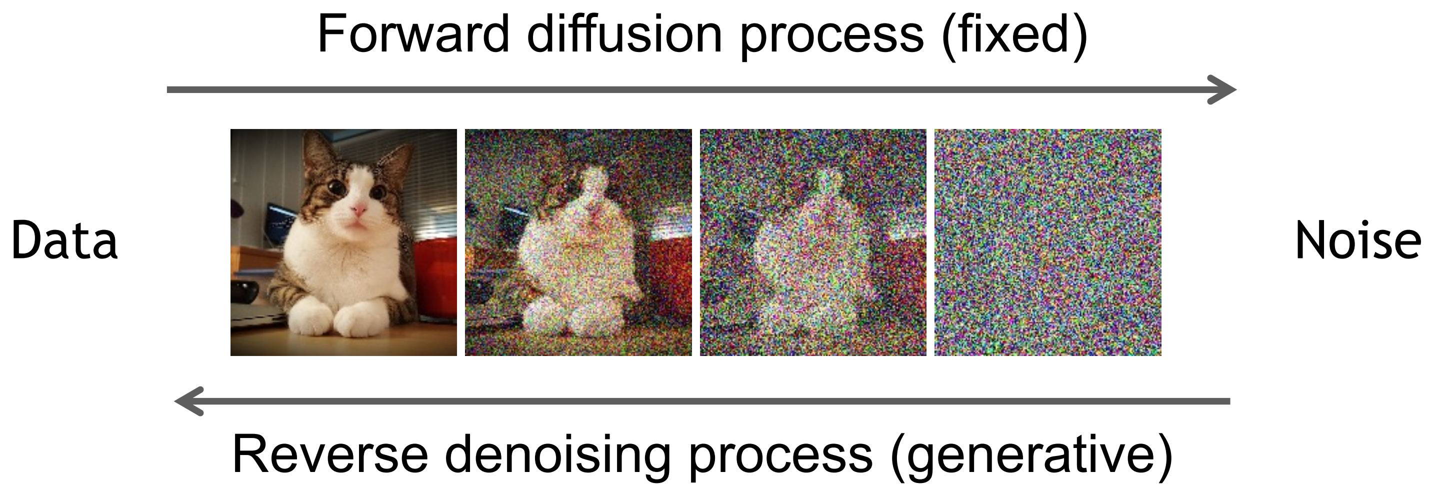
Sohl-Dickstein, ICML 2015
Ho et al., NeurIPS 2020

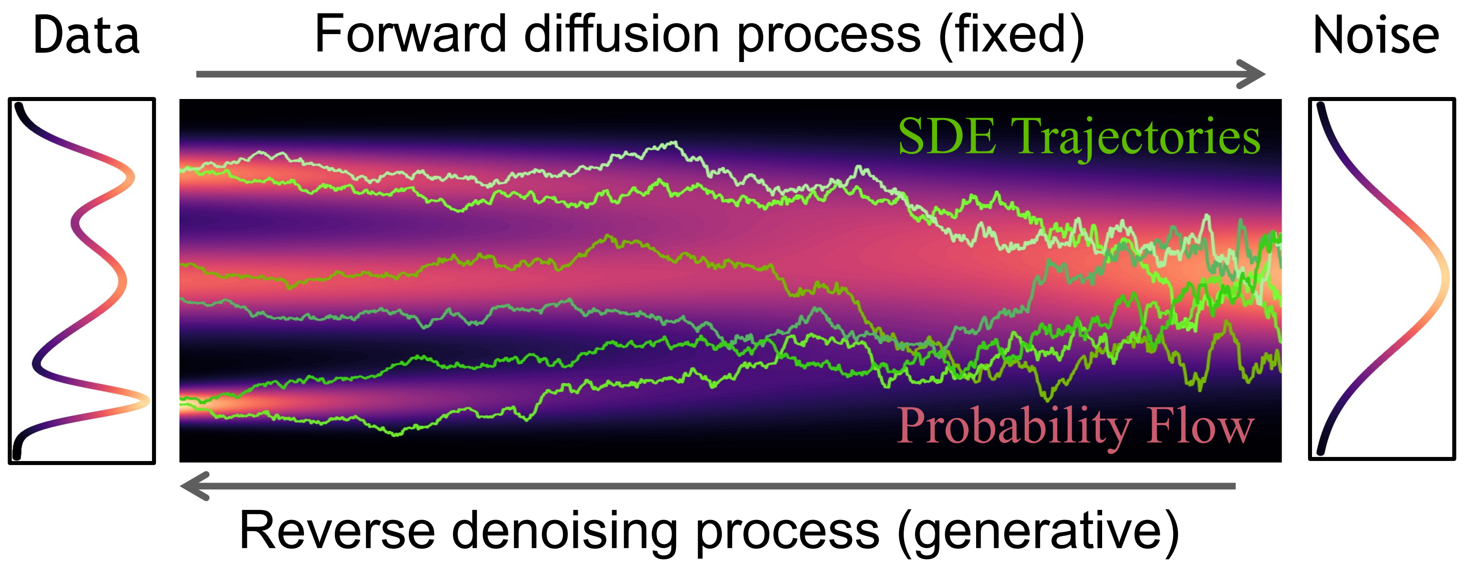
DIFFICULTY OF TRAINING DDMS IN DATA SPACE

The generative process in DDMs can be described by stochastic differential equations (SDEs) as shown by Song et al. ICLR 2021.

Given a highly complex and multimodal data distribution:

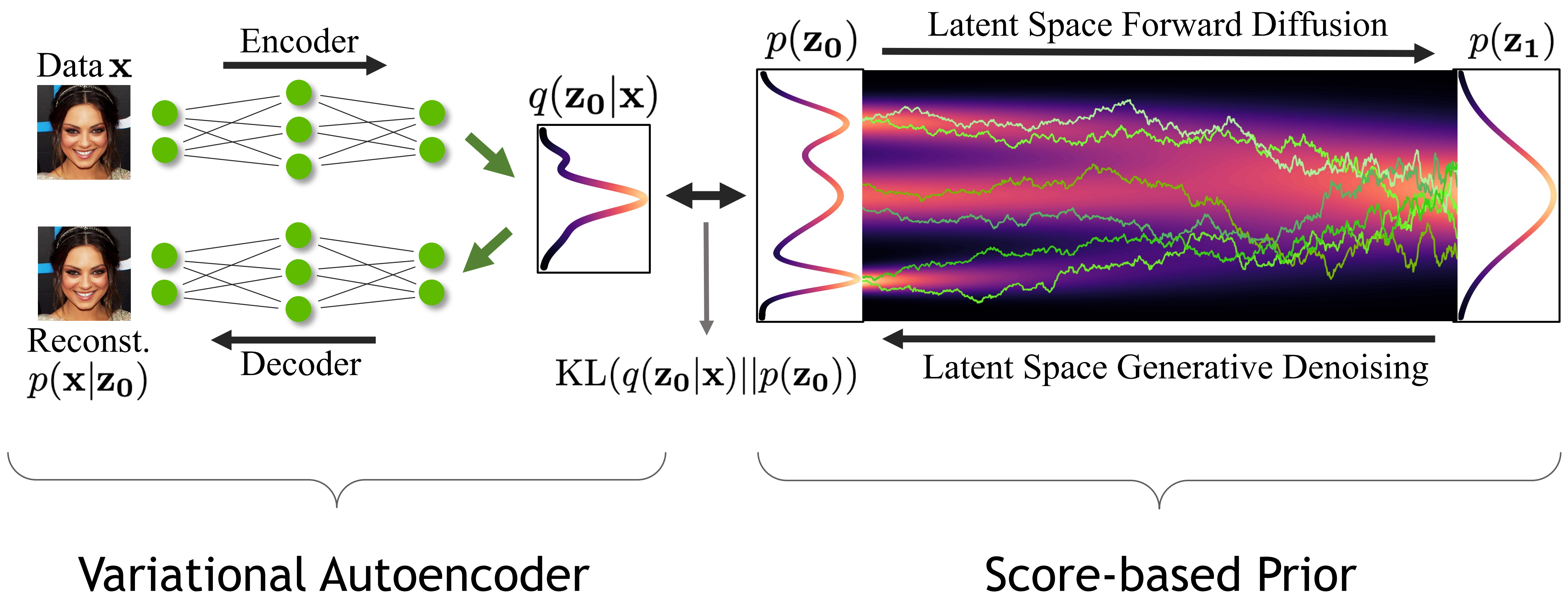
- The time evolution in generative SDEs is complex.
- Numerical solves require 1000s of steps.
- DDMs can be only applied to continuous data.





SCORE-BASED GENERATIVE MODELING IN LATENT SPACE

Variational Autoencoder + Score-based Prior



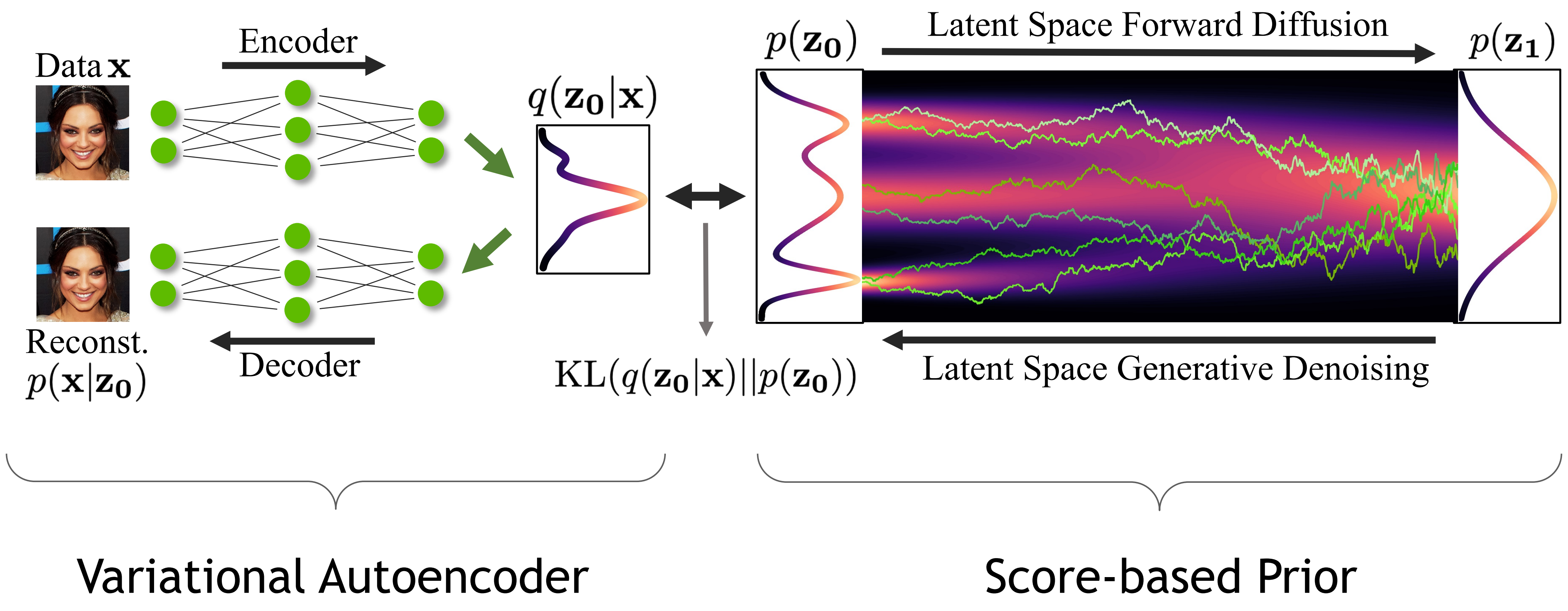
Main Idea

Encoder maps the input data to an embedding space

Score-based generative model is applied in the latent space

SCORE-BASED GENERATIVE MODELING IN LATENT SPACE

Variational Autoencoder + Score-based Prior



Advantages:

(1) The distribution of latent embeddings close to Normal distribution → *Simpler denoising, Faster Synthesis!*

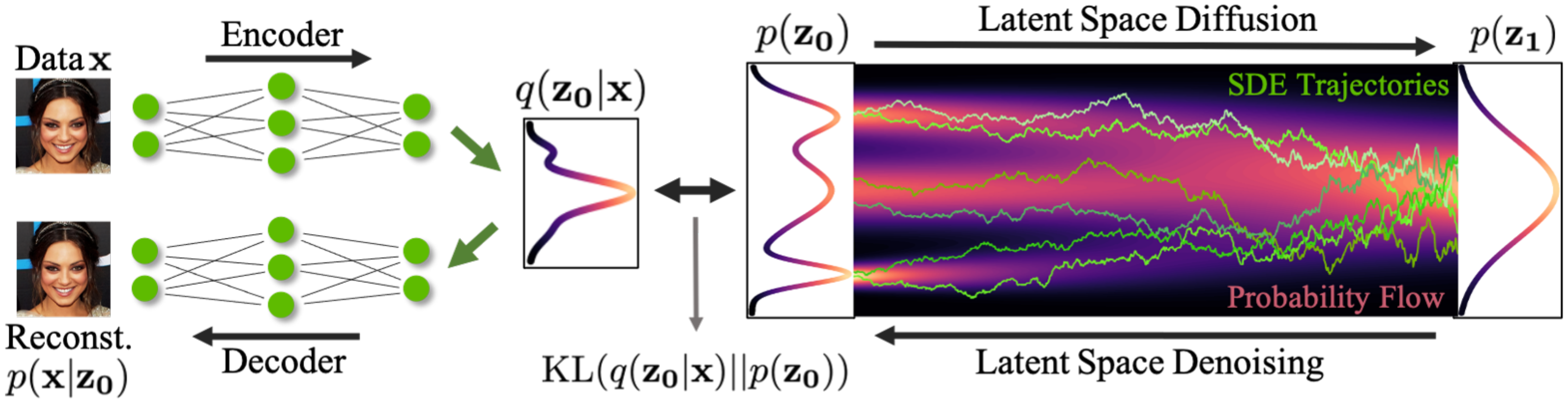
(2) Augmented latent space → *More expressivity!*

(3) Tailored Autoencoders → *More expressivity, Application to any data type (graphs, text, 3D data, etc.) !*

TECHNICAL CONTRIBUTIONS

TRAINING OBJECTIVE

Score matching for the cross entropy



$$\mathcal{L}(\mathbf{x}, \boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{\psi}) = \mathbb{E}_{q_{\phi}(\mathbf{z}_0|\mathbf{x})}\left[-\log p_{\psi}(\mathbf{x}|\mathbf{z}_0)\right] + \mathrm{KL}(q_{\phi}(\mathbf{z}_0|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}_0))$$

$$= \underbrace{\mathbb{E}_{q_{\phi}(\mathbf{z}_0|\mathbf{x})}\left[-\log p_{\psi}(\mathbf{x}|\mathbf{z}_0)\right]}_{\text{reconstruction term}} + \underbrace{\mathbb{E}_{q_{\phi}(\mathbf{z}_0|\mathbf{x})}\left[\log q_{\phi}(\mathbf{z}_0|\mathbf{x})\right]}_{\text{negative encoder entropy}} + \underbrace{\mathbb{E}_{q_{\phi}(\mathbf{z}_0|\mathbf{x})}\left[-\log p_{\boldsymbol{\theta}}(\mathbf{z}_0)\right]}_{\text{cross entropy}}$$

$$CE(q(\mathbf{z}_0|\mathbf{x})||p(\mathbf{z}_0)) = \underbrace{\mathbb{E}_{t\sim\mathcal{U}[0,1]}}_{\text{time sampling}}\left[\frac{g(t)^2}{2}\underbrace{\mathbb{E}_{q(\mathbf{z}_t,\mathbf{z}_0|\mathbf{x})}}_{\text{Forward diffusion}}\left[||\underbrace{\nabla_{\mathbf{z}_t}\log q(\mathbf{z}_t|\mathbf{z}_0)}_{\text{Diffusion kernel}} - \underbrace{\nabla_{\mathbf{z}_t}\log p(\mathbf{z}_t)}_{\text{Trainable score function}}||_2^2\right]\right] + \underbrace{\frac{D}{2}\log\left(2\pi e\sigma_0^2\right)}_{\text{Constant}}$$

NORMAL DISTRIBUTION ASSUMPTION

Inductive biases for training

Recall that the distribution of latent variables is close to a Normal distribution:

$$CE(q(\mathbf{z}_0|\mathbf{x})||p(\mathbf{z}_0)) = \mathbb{E}_{t\sim\mathcal{U}[0,1]}\left[\frac{g(t)^2}{2}\mathbb{E}_{q(\mathbf{z}_t,\mathbf{z}_0|\mathbf{x})}\left[||\nabla_{\mathbf{z}_t}\log q(\mathbf{z}_t|\mathbf{z}_0) - \nabla_{\mathbf{z}_t}\log p(\mathbf{z}_t)||_2^2\right]\right]$$

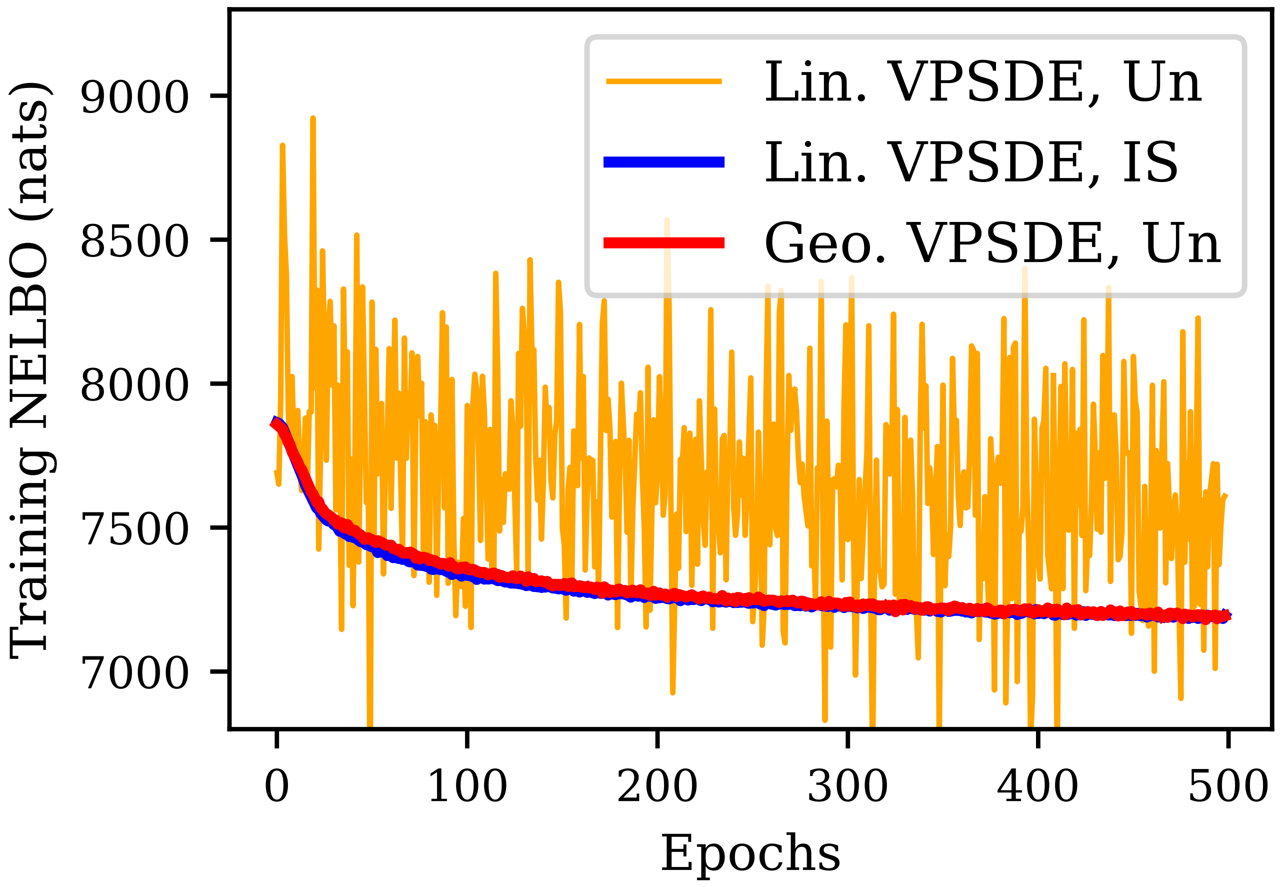
Define variance reduction techniques using importance sampling for the Normal assumption

Design score function that is close to a Normal score function

NORMAL DISTRIBUTION ASSUMPTION

Inductive biases for training

Recall that the distribution of latent variables is close to a Normal distribution:



EXPERIMENTAL RESULTS

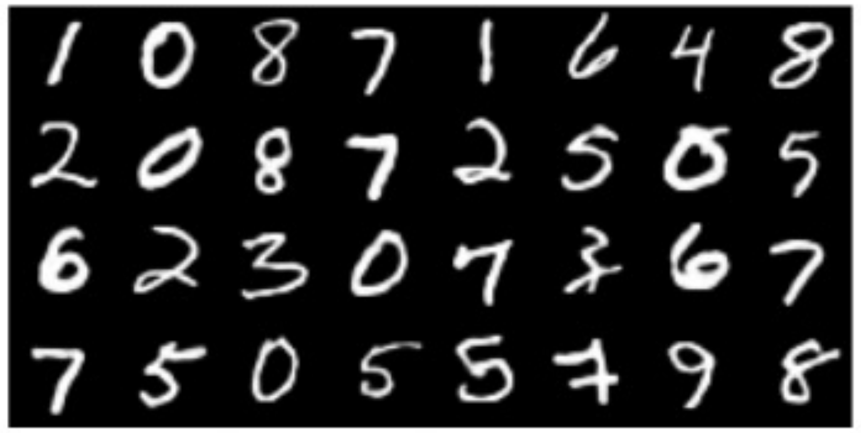
EXPERIMENTAL RESULTS (1)

Model samples

(a) CIFAR-10

(b) CelebA-HQ-256

(c) OMNIGLOT

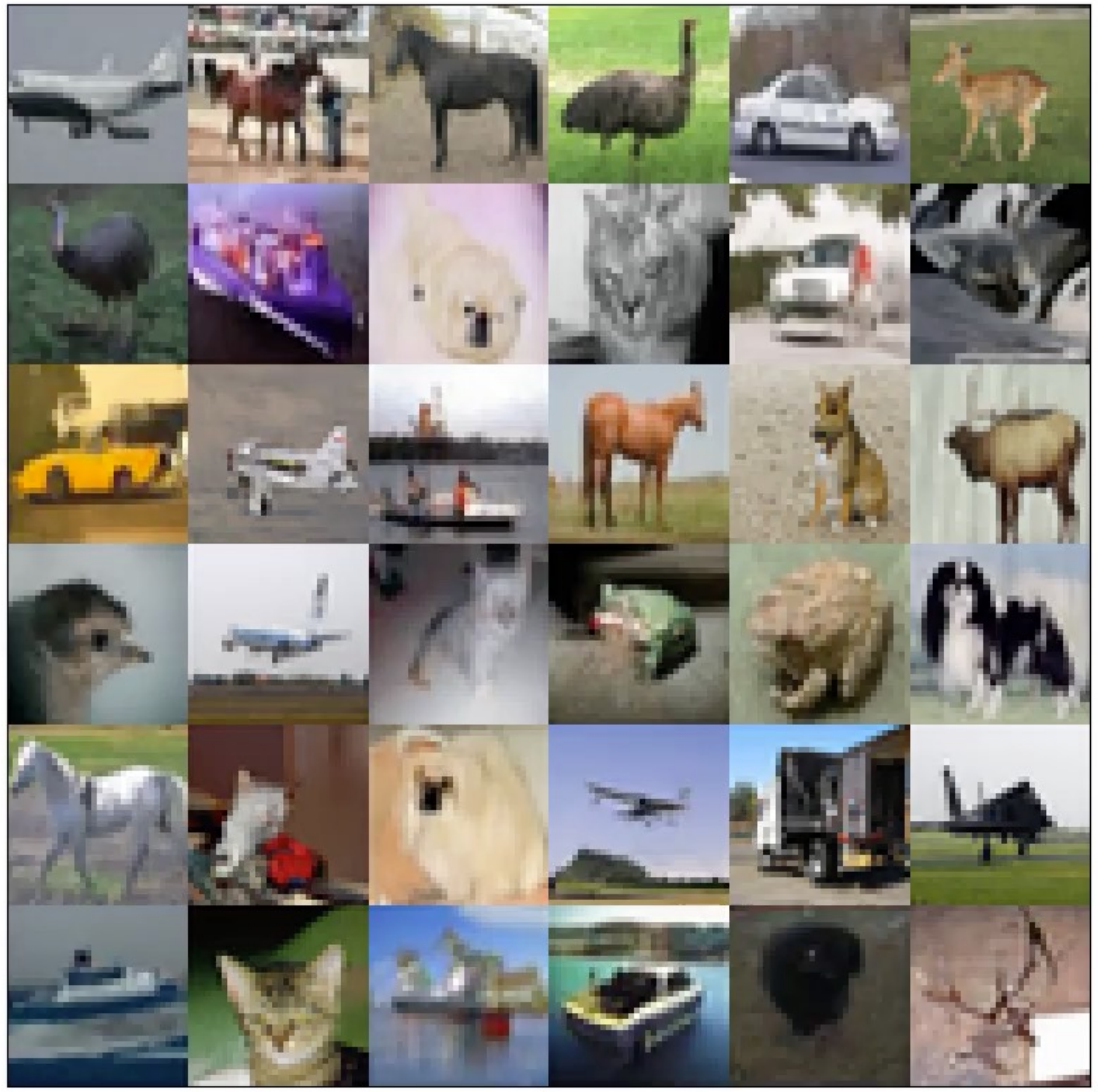


(d) MNIST

EXPERIMENTAL RESULTS (3)

Latent space interpolations

CIFAR-10



CelebA-HQ-256

EVOLUTION OF SAMPLES IN LATENT SPACE

Latent samples fed to decoder

EXPERIMENTAL RESULTS (4)

Sampling speed on CelebA-HQ 256

Pixel-space Score-based Model (Song et al., ICLR 2021):

- “Predictor-Corrector” sampling: 4000 network calls, ~45 min.

LSGM (ours):

- Adaptive ODE-sampler: 23 network calls, ~4 sec., better quality

→ ***~675x faster sampling!***

1. Low spatial dimension in latent space (32x32) → we use shallower network for sufficient receptive field.
2. Marginal posterior already close to Normal → smooth SDE/ODE, numerically fast to solve.
3. Decoder can “correct” small errors from ODE solve. No direct pixel space artifacts.

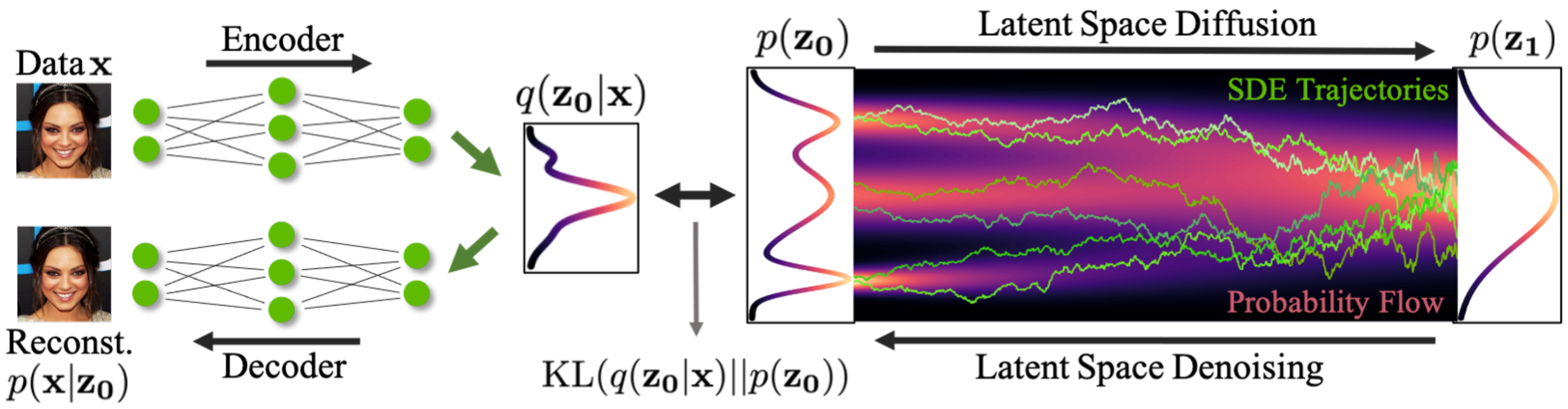
SUMMARY

We propose the latent score-based generative model (LSGM)

LSGM embeds the data into a smooth latent space and models the distribution over encodings using a score-based prior. This has multiple advantages:

- **Increased model expressivity** (due to latent variables and additional encoder and decoder)
- **Increased synthesis speed** (due to smooth latent space distribution)
- **Increased data type flexibility** (encoder and decoder can be tailored to data type)

https://nvlabs.github.io/LSGM/



NVIDIA

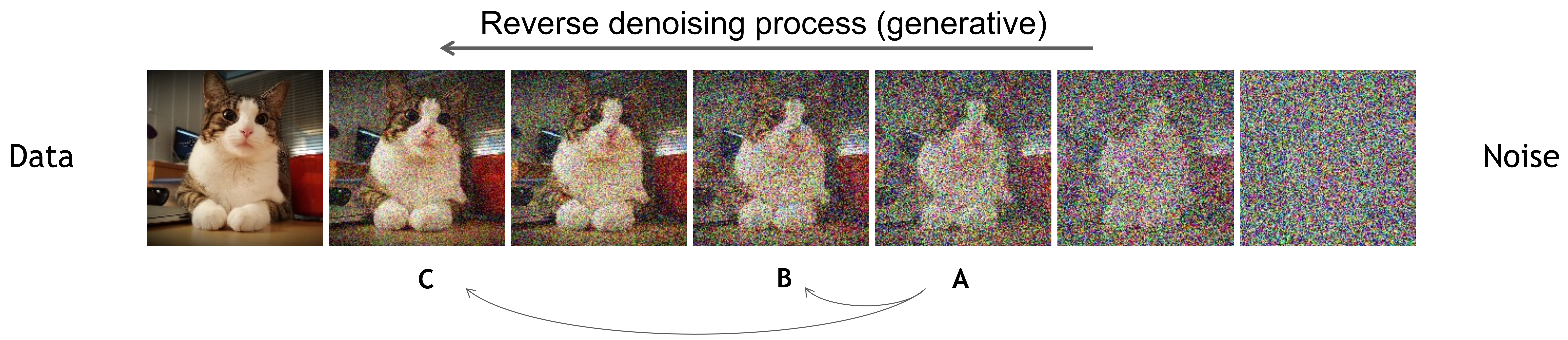
Denoising Diffusion GANs

Zhisheng Xiao, Karsten Kreis, Arash Vahdat

ICLR 2021 (spotlight)

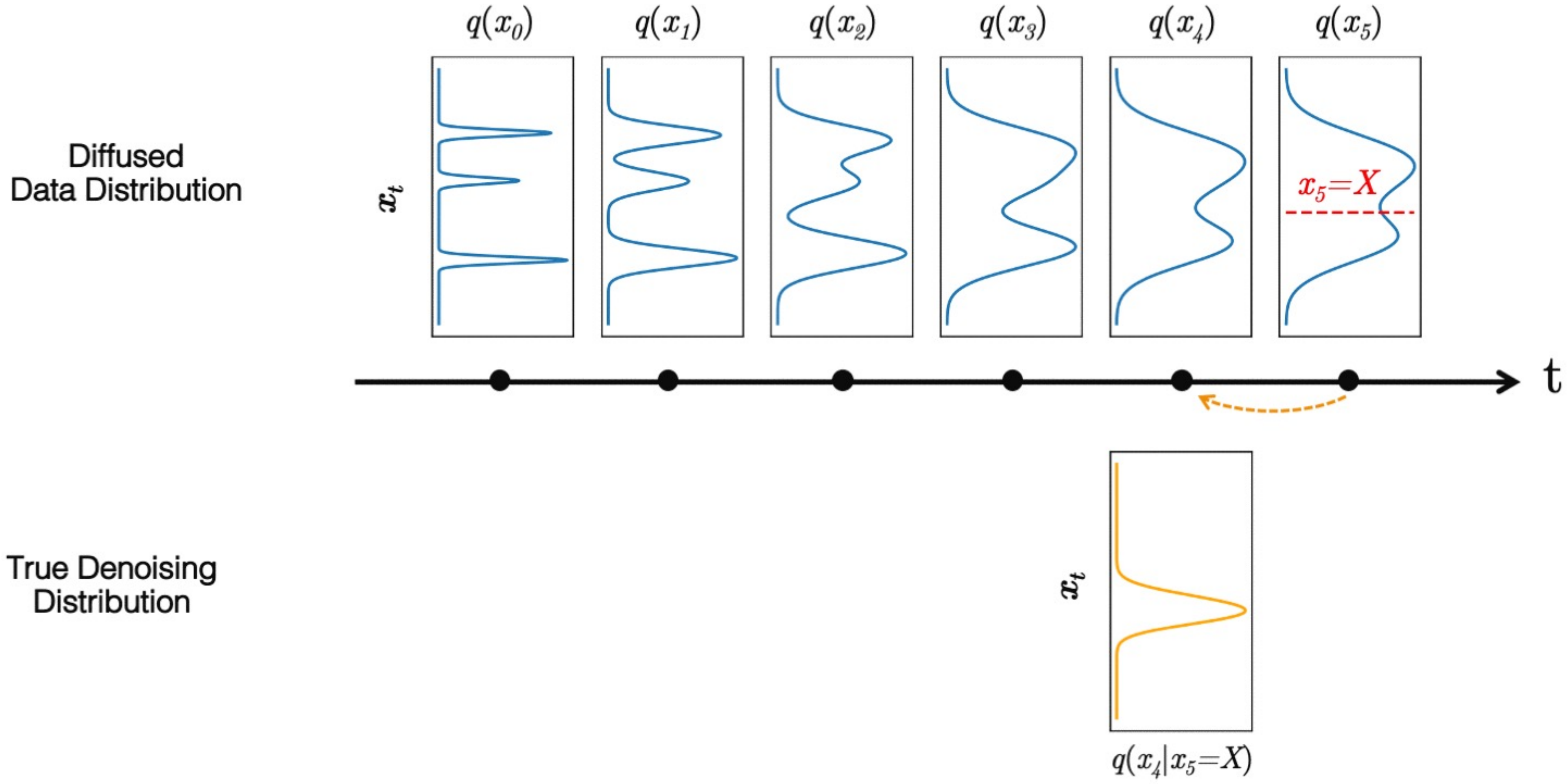
LARGE STEP DENOISING DISTRIBUTION

- The main idea of LSGM is to bring the distribution of data as close as possible to the Normal distribution.
- What if we don't change the data distribution and try denoising for large steps:



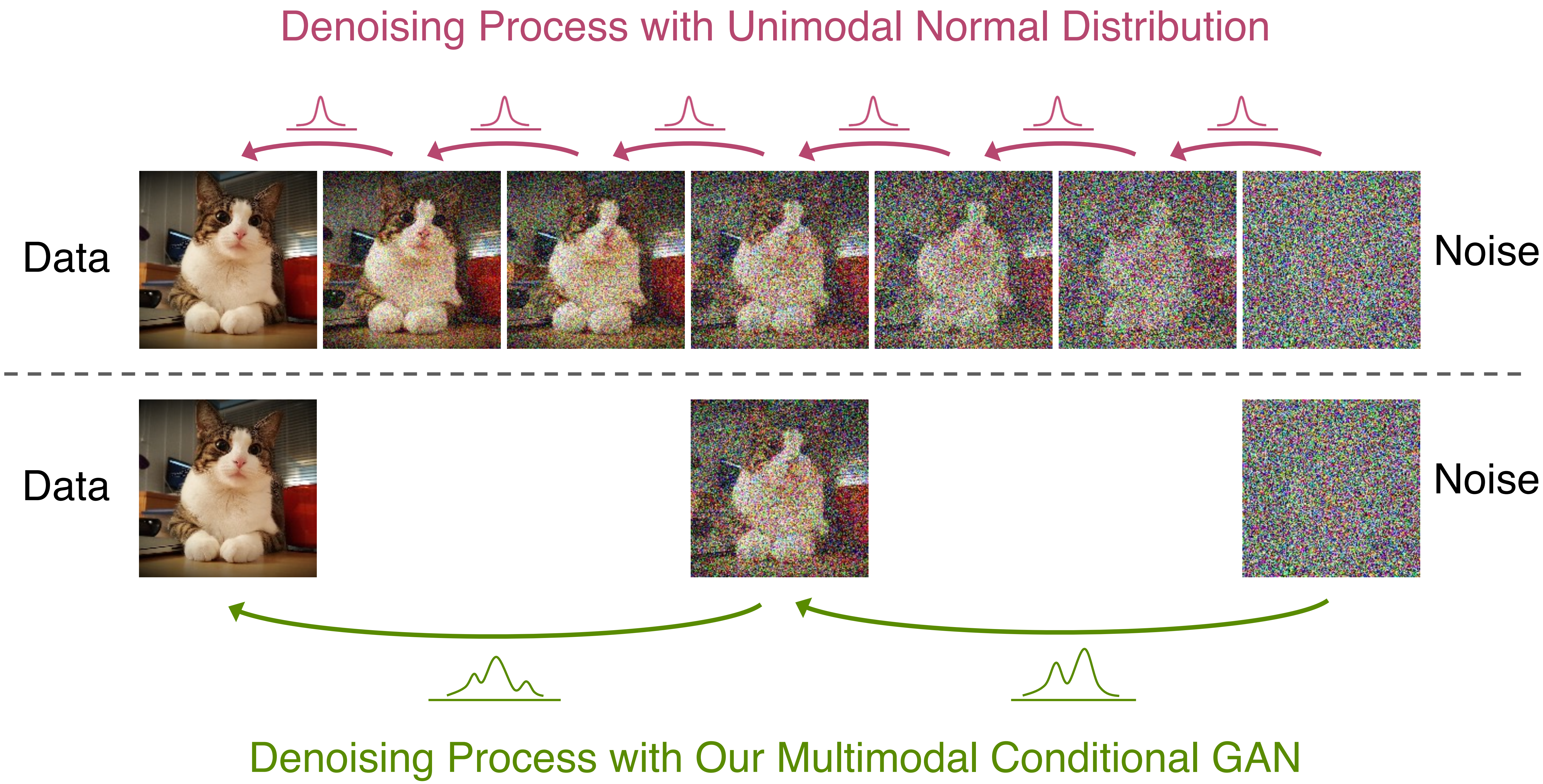
NORMAL ASSUMPTION IN DENOISING DISTRIBUTION

Holds for small steps



NORMAL ASSUMPTION IN DENOISING DISTRIBUTION

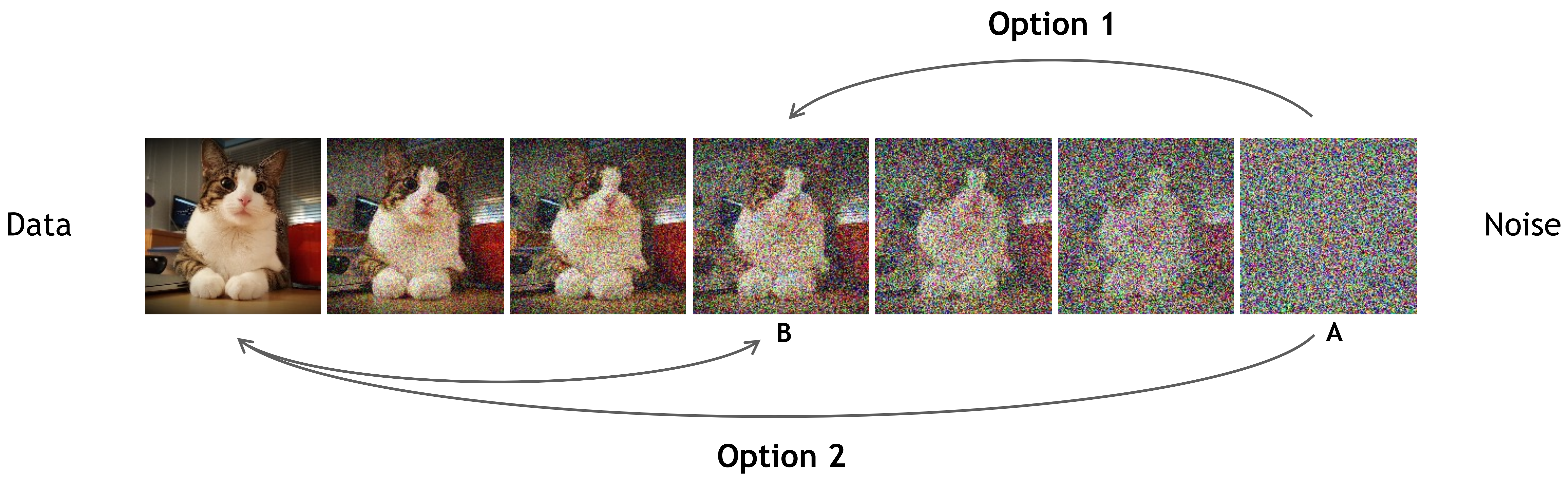
Holds for small steps



DENOISING DIFFUSION GAN

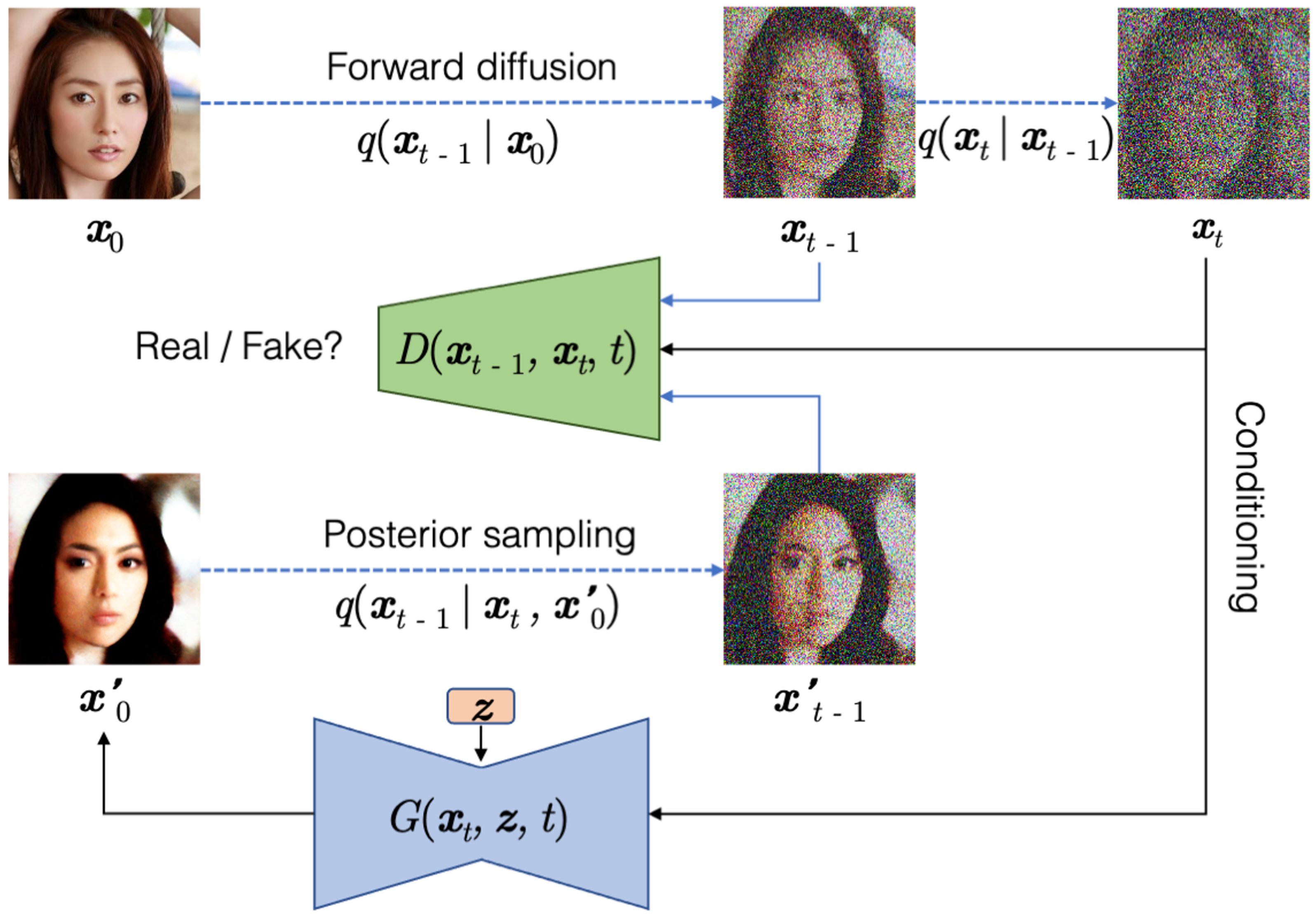
Parameterization

- How can we parameterize the conditional GAN generator?



ADVERSARIAL TRAINING

- How can we train the conditional GAN generator: $$\min_{\theta} \sum_{t \geq 1} \mathbb{E}_{q(\mathbf{x}_t)} \left[D_{\text{adv}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \right]$$



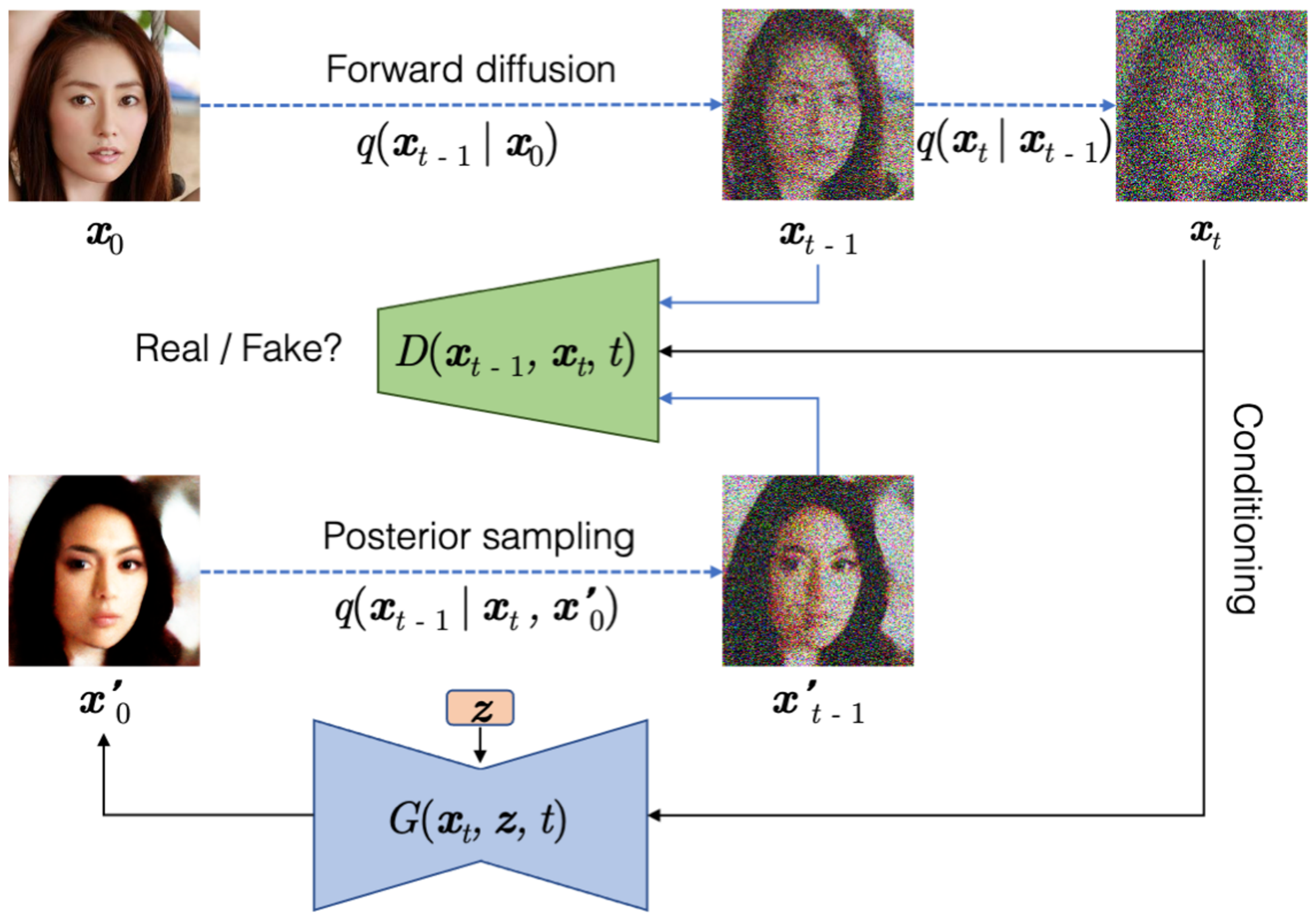
ADVANTAGES OVER TRADITIONAL GANS

Why not to train a one-shot GAN generator:

- Stronger mode coverage
- Better training stability

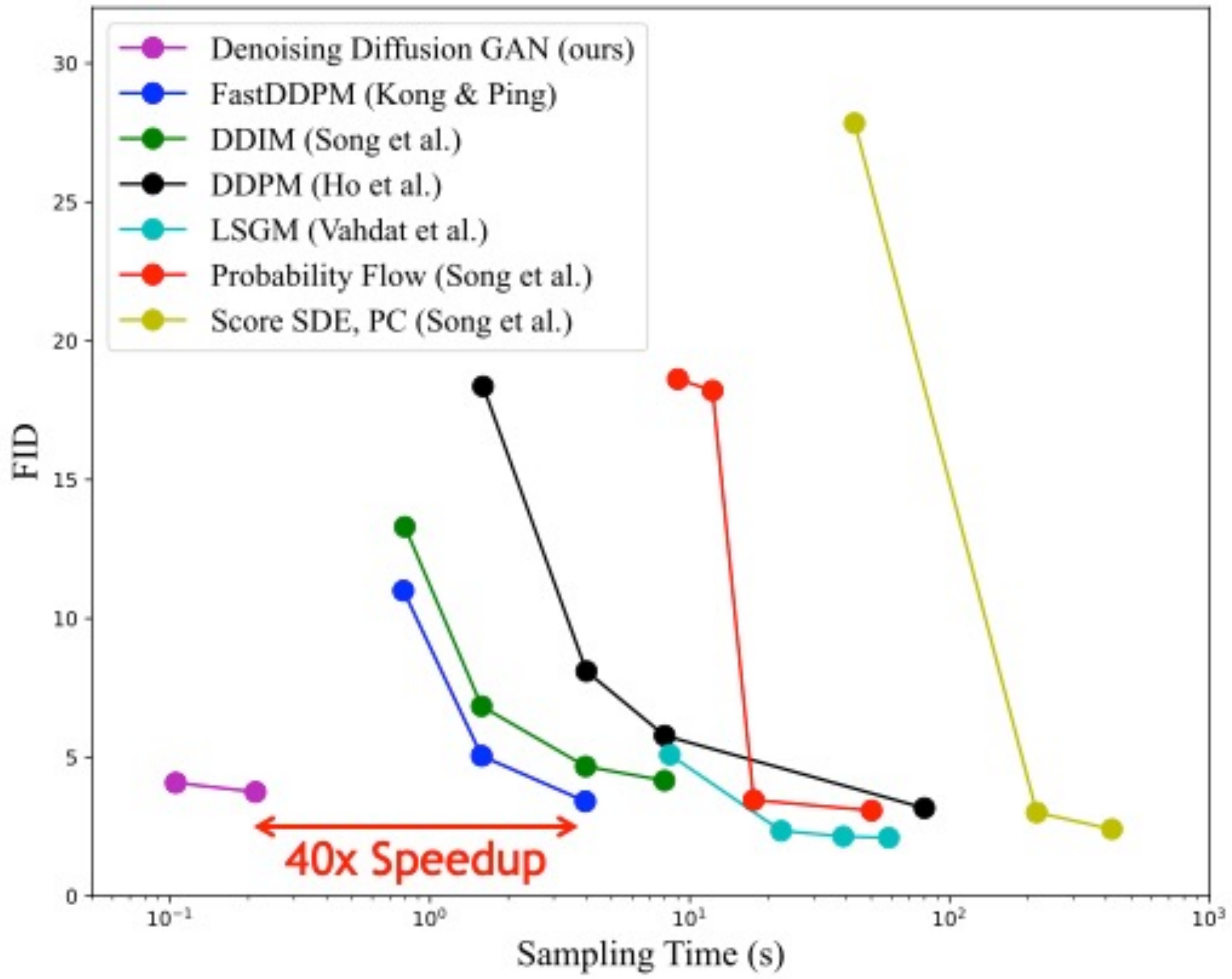
Both generator and discriminator are solving a much simpler problem.

$$\min_{\theta} \sum_{t\geq 1} \mathbb{E}_{q(\mathbf{x}_t)} \left[D_{\mathrm{adv}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \right]$$

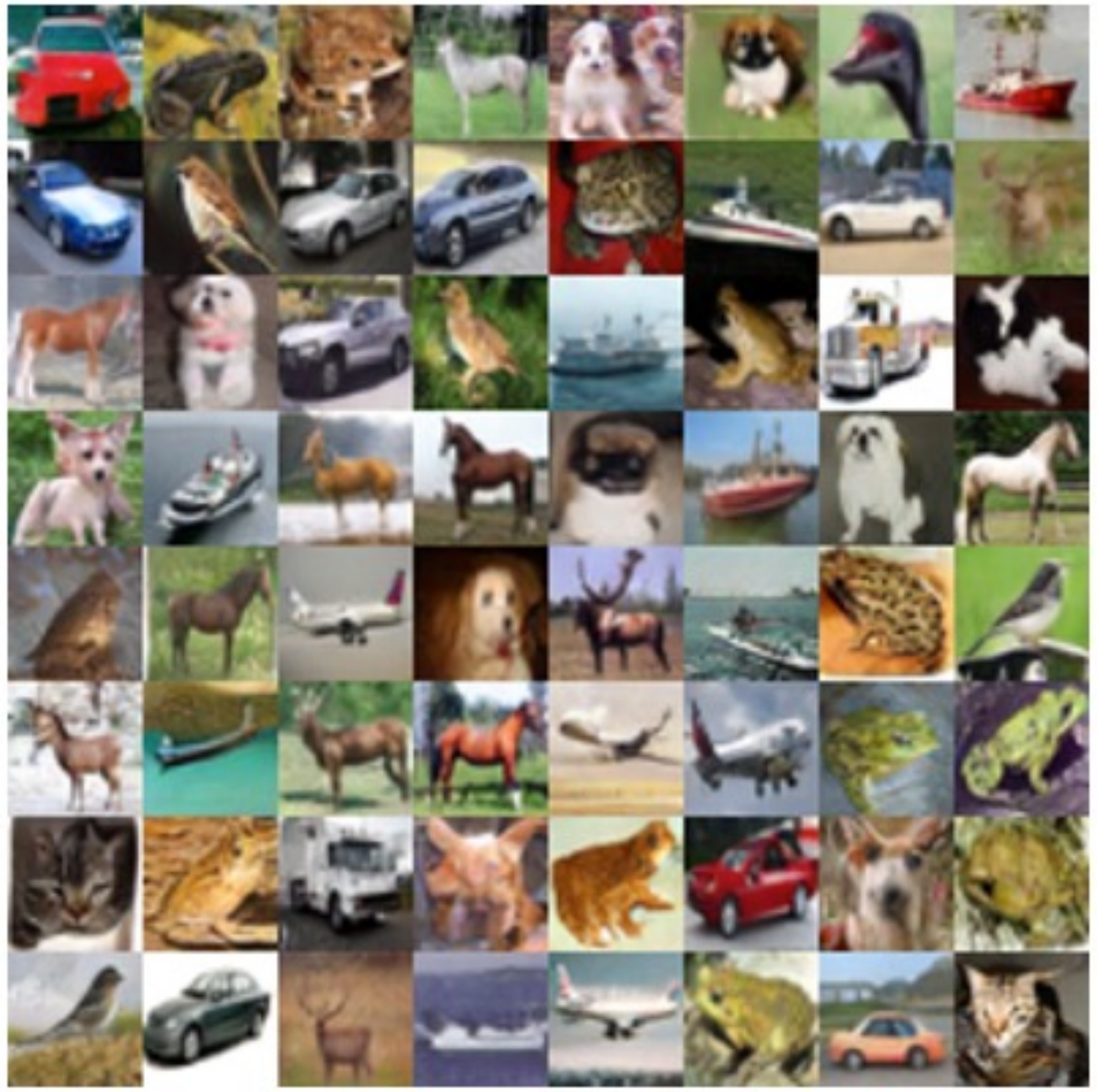


EXPERIMENTAL RESULTS

CIFAR-10 dataset



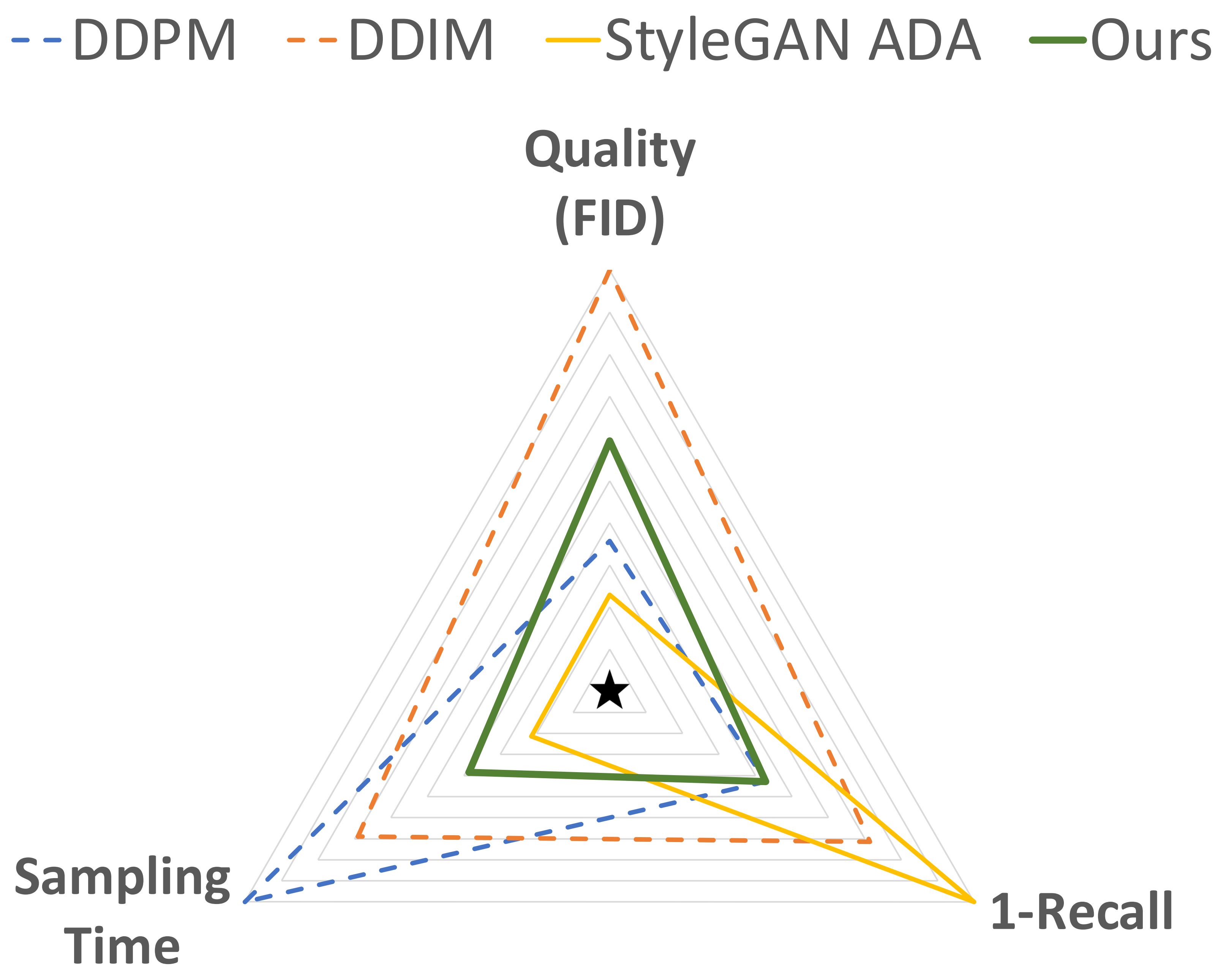
Sample Quality vs. Sampling Time



CIFAR-10 Samples

RESULTS ON THE GENERATIVE LEARNING TRILEMMA

CIFAR-10 dataset



OTHER DATASETS

CelebA-HQ 256

LSUN Churches Outdoor 256

SAMPLING TIME

CelebA-HQ 256

Model	FID ↓	# Fun. Calls
Song et al. ICLR 2021	7.22	4000
Latent Space Diffusion Models, NeurIPS 2021	7.23	23
Denoising Diffusion GANs, ICLR 2022	7.60	2

SUMMARY

Denoising diffusion GANs

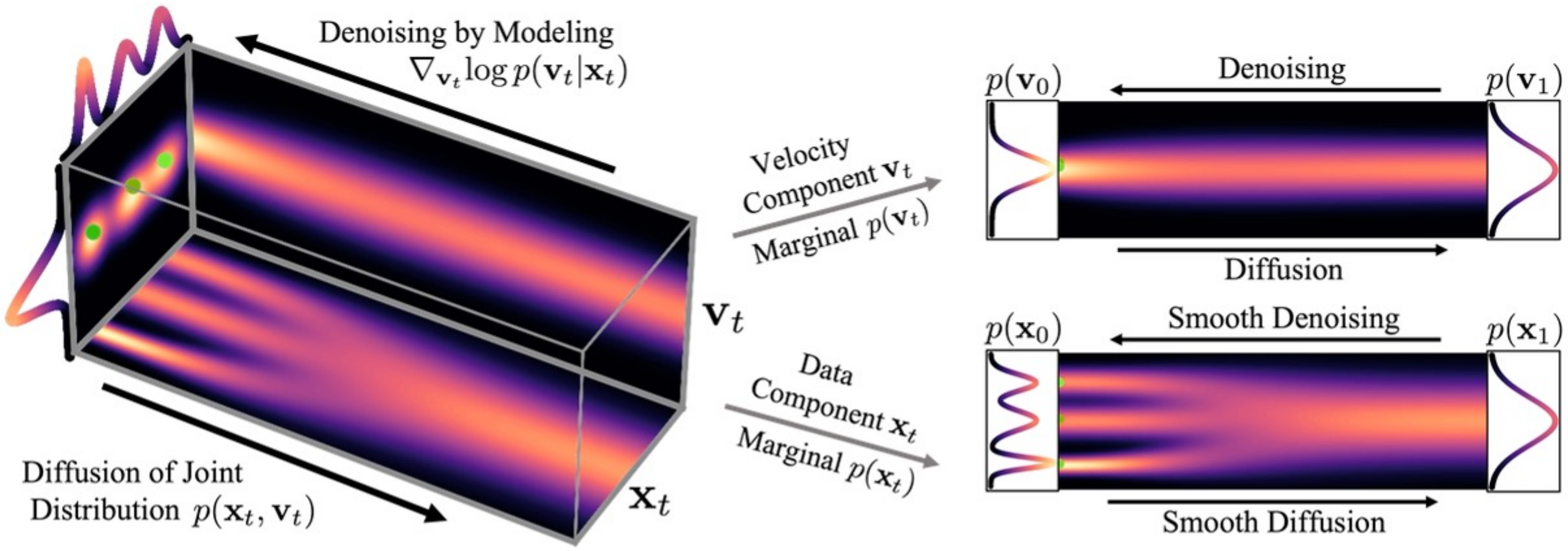
We introduce denoising diffusion GANs to tackle the generative learning trilemma:

- **Faster sampling:** due to multimodal complex denoising distribution
- **Better mode coverage:** due to simple generation problem at each step
- **High-quality samples:** due to the adversarial training

https://nvlabs.github.io/denoising-diffusion-gan/

WHAT’S NEXT?

Score-Based Generative Modeling with Critically-Damped Langevin Diffusion



Forward
Diffusion Process

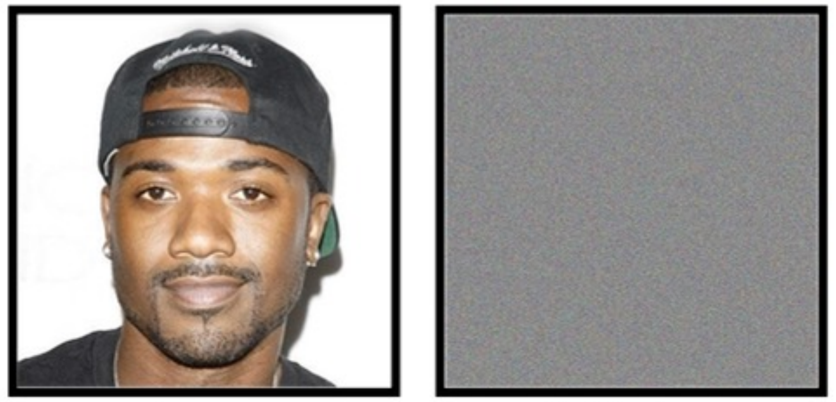


Image $\mathbf{x}_t$ Velocity $\mathbf{v}_t$

$t = 0.00$

Karsten Kreis’s Talk on Feb 17

THANK YOU!