Diffusion Probabilistic Models: Theory and Applications

Fan Bao

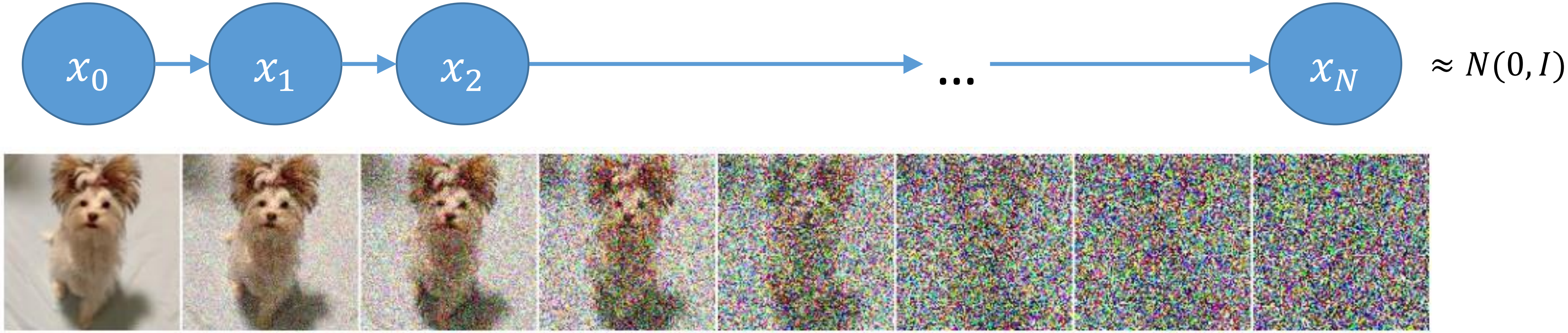
Tsinghua University

Diffusion Probabilistic Models (DPMs)

Ho et al. Denoising diffusion probabilistic models (DDPM), Neurips 2020.
Song et al. Score-based generative modeling through stochastic differential equations, ICLR 2021.
Bao et al. Analytic-DPM: an Analytic Estimate of the Optimal Reverse Variance in Diffusion Probabilistic Models, ICLR 2022.
Bao et al. Estimating the Optimal Covariance with Imperfect Mean in Diffusion Probabilistic Models, ICML 2022.

- Diffusion process gradually injects noise to data
- Described by a Markov chain: $q(x_0, \dots, x_N) = q(x_0)q(x_1|x_0) \dots q(x_N|x_{N-1})$

Transition of diffusion: $q(x_n|x_{n-1}) = N(\sqrt{\alpha_n}x_{n-1}, \beta_n I)$ $\quad \alpha_n = 1 - \beta_n$

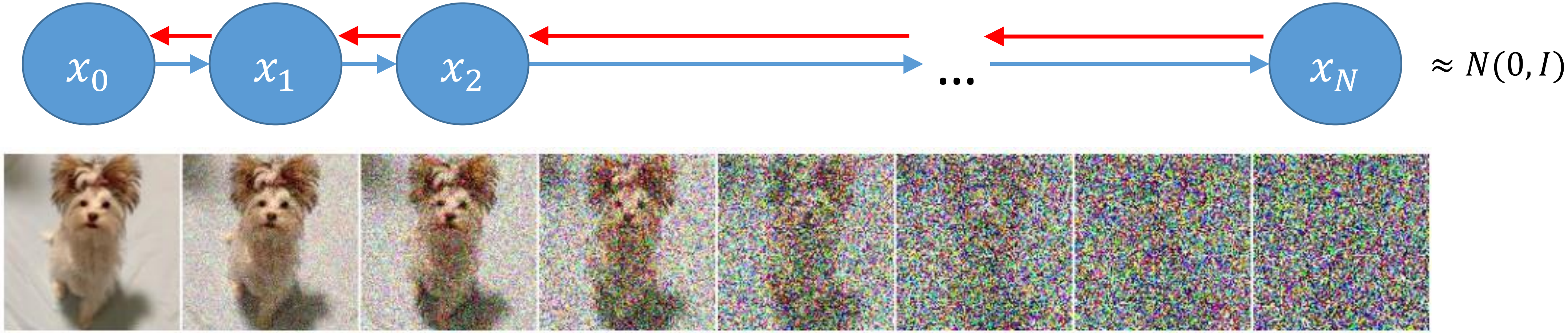


Diffusion process: $q(x_0, \dots, x_N) = q(x_0)q(x_1|x_0) \dots q(x_N|x_{N-1})$

Demo Images from *Song et al. Score-based generative modeling through stochastic differential equations, ICLR 2021.*

- Diffusion process in the reverse direction $\Leftrightarrow$ denoising process
- Reverse factorization: $q(x_0, \ldots, x_N) = q(x_0|x_1) \ldots q(x_{N-1}|x_N) q(x_N)$

Transition of denoising: $q(x_{n-1}|x_n) = ?$



Diffusion process: $q(x_0, \ldots, x_N) = q(x_0) q(x_1|x_0) \ldots q(x_N|x_{N-1})$

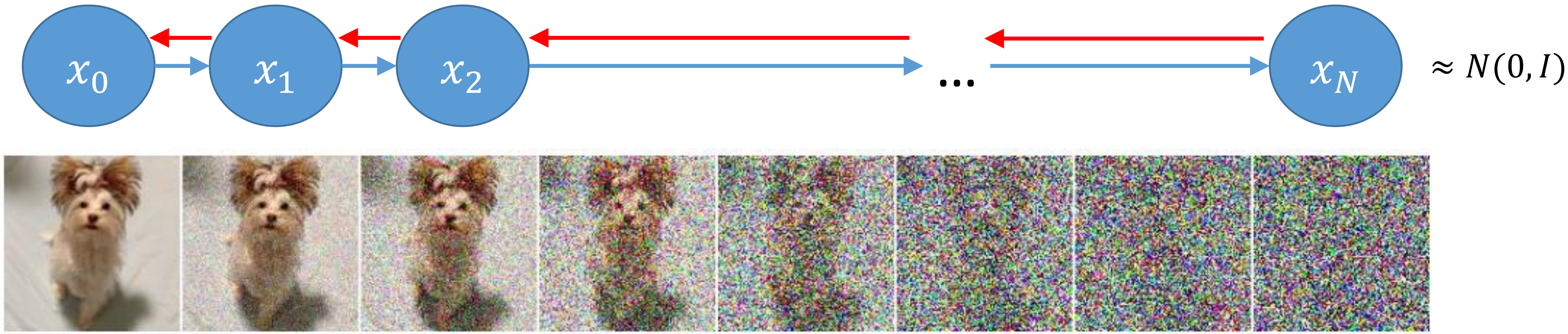
$= q(x_0|x_1) \ldots q(x_{N-1}|x_N) q(x_N)$

- Approximate diffusion process in the reverse direction

Model transition: $p(x_{n-1}|x_n) = N(\mu_n(x_n), \Sigma_n(x_n))$

$\downarrow$ approximate

Transition of denoising: $q(x_{n-1}|x_n) = ?$



Diffusion process: $q(x_0, \dots, x_N) = q(x_0)q(x_1|x_0) \dots q(x_N|x_{N-1})$

$= q(x_0|x_1) \dots q(x_{N-1}|x_N)q(x_N)$

The model: $p(x_0, \dots, x_N) = p(x_0|x_1) \dots p(x_{N-1}|x_N)p(x_N)$

- We hope $q(x_0, \dots, x_N) \approx p(x_0, \dots, x_N)$ $\quad p(x_{n-1}|x_n) = N(\mu_n(x_n), \Sigma_n(x_n))$
- Achieved by minimizing their KL divergence (i.e., maximizing the ELBO)

min KL $\quad\quad$ max ELBO

$$\min_{\mu_n, \Sigma_n} KL(q(x_{0:N})||p(x_{0:N})) \Leftrightarrow \max_{\mu_n, \Sigma_n} \mathrm{E}_q \log \frac{p(x_{0:N})}{q(x_{1:N}|x_0)}$$

What is the optimal solution?

Bao et al. Analytic-DPM: an Analytic Estimate of the Optimal Reverse Variance in Diffusion Probabilistic Models, ICLR 2022.

Theorem (*The optimal solution under scalar variance, i.e.,* $\Sigma_n(x_n) = \sigma_n^2 I$)

The optimal solution to $\min_{\mu_n(\cdot),\sigma_n^2} KL(q(x_{0:N})||p(x_{0:N}))$ is

$$\mu_n^*(x_n) = \frac{1}{\sqrt{\alpha_n}}(x_n + \beta_n \nabla \log q_n(x_n)),$$

$$\sigma_n^{*2} = \frac{\beta_n}{\alpha_n}(1 - \beta_n \mathrm{E}_{q_n(x_n)} \frac{\|\nabla \log q_n(x_n)\|^2}{d}).$$

3 key steps in proof:

- Moment matching
- Law of total variance
- Score representation of moments of $q(x_0|x_n)$

Noise prediction form:

$\nabla \log q_n(x_n) = -\frac{1}{\sqrt{\bar{\beta}_n}} \mathrm{E}_{q(x_0|x_n)}[\epsilon_n]$

Estimated by predicting noise

Parameterization of $\boldsymbol{\mu_n}(\cdot)$:

$$\mu_n(x_n) = \frac{1}{\sqrt{\alpha_n}}\left(x_n - \beta_n \frac{1}{\sqrt{\bar{\beta}_n}} \hat{\epsilon}_n(x_n)\right)$$

Bao et al. Estimating the Optimal Covariance with Imperfect Mean in Diffusion Probabilistic Models, ICML 2022.

Theorem (*The optimal solution for diagonal covariance , i.e.,* $\Sigma_n(x_n) = \mathrm{diag}(\sigma_n(x_n)^2)$)

The optimal solution to $\min_{\mu_n(\cdot),\sigma_n(\cdot)^2} KL(q(x_{0:N})||p(x_{0:N}))$ is

$$\mu_n^*(x_n) = \frac{1}{\sqrt{\alpha_n}}(x_n + \beta_n \boxed{\nabla \log q_n(x_n)}),$$

Predict noise

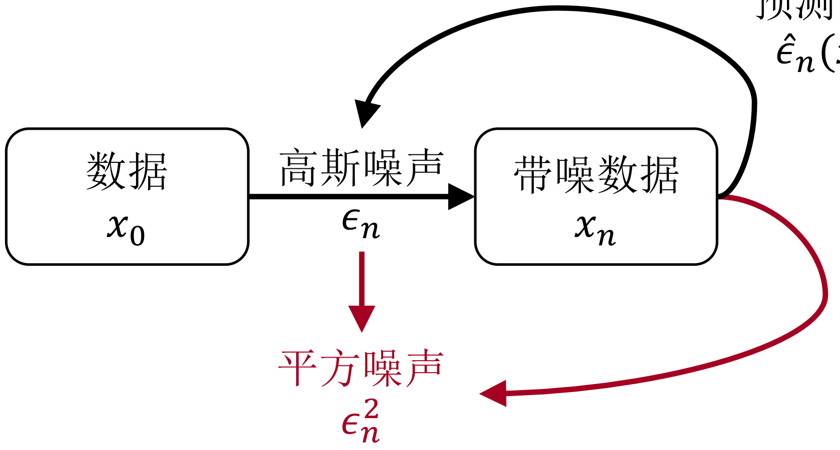
$$\sigma_n^*(x_n)^2 = \frac{\overline{\beta}_{n-1}}{\overline{\beta}_n}\beta_n + \frac{\beta_n^2}{\overline{\beta}_n \alpha_n}(\boxed{\mathrm{E}_{q(x_n|x_n)}[\epsilon_n^2]} - \boxed{\mathrm{E}_{q(x_n|x_n)}[\epsilon_n]}^2).$$

Predict squared noise

- Implementation framework of predicting squared noise

最优协方差表达式: $\boldsymbol{\sigma}_n^*(\boldsymbol{x}_n)^2 = \lambda_n^2 \mathbf{1} + \gamma_n^2 \frac{\overline{\beta}_n}{\overline{\alpha}_n}\left(\boxed{\mathbb{E}_{q(\boldsymbol{x}_0|\boldsymbol{x}_n)}[\boldsymbol{\epsilon}_n^2]} - \boxed{\mathbb{E}_{q(\boldsymbol{x}_0|\boldsymbol{x}_n)}[\boldsymbol{\epsilon}_n]}^2\right)$

constant



预测网络 $\hat{\epsilon}_n(x_n)$

最小化均方误差 $\min_{\hat{\epsilon}_n} \mathbf{E}\|\hat{\epsilon}_n(x_n) - \epsilon_n\|_2^2$

预测网络 $h_n(x_n)$

最小化均方误差 $\min_{h_n} \mathbf{E}\|h_n(x_n) - \epsilon_n^2\|_2^2$

基于预测噪声平方的最优协方差估计: $\hat{\boldsymbol{\sigma}}_n(\boldsymbol{x}_n)^2 = \lambda_n^2 \mathbf{1} + \gamma_n^2 \frac{\overline{\beta}_n}{\overline{\alpha}_n}\left(\boxed{\boldsymbol{h}_n(\boldsymbol{x}_n)} - \hat{\boldsymbol{\epsilon}}_n(\boldsymbol{x}_n)^2\right).$

Bao et al. Estimating the Optimal Covariance with Imperfect Mean in Diffusion Probabilistic Models, ICML 2022.

Theorem (*The optimal solution for diagonal covariance , i.e.,* $\Sigma_n(x_n) = \mathrm{diag}(\sigma_n(x_n)^2)$)

The optimal solution to $\min_{\sigma_n(\cdot)^2} KL(q(x_{0:N})||p(x_{0:N}))$ with imperfect mean is

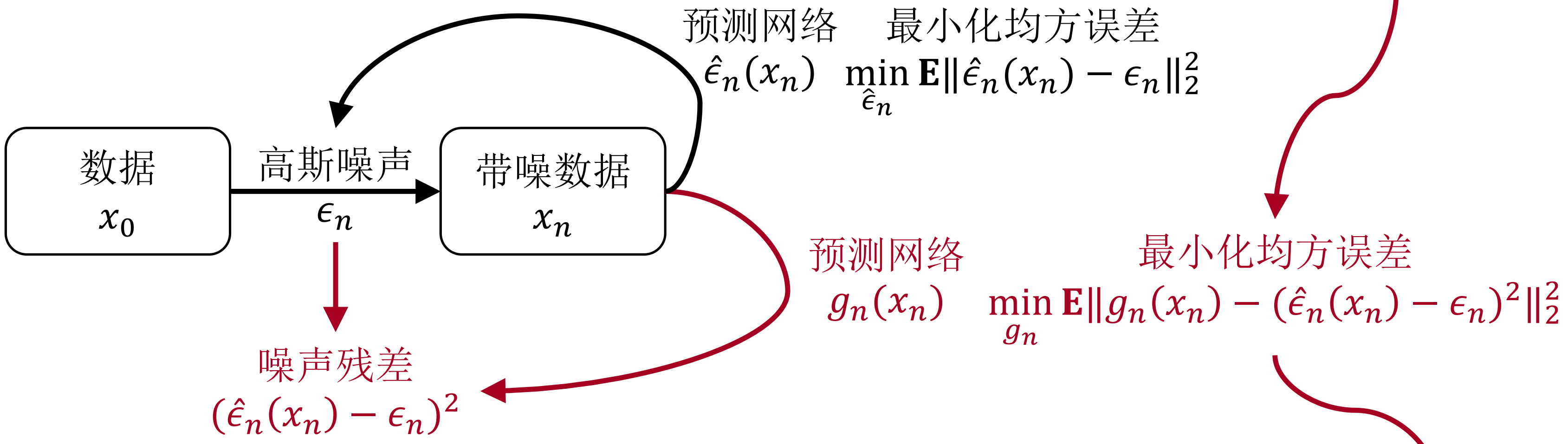
$$\tilde{\sigma}_n^*(x_n)^2 = \frac{\overline{\beta}_{n-1}}{\overline{\beta}_n}\beta_n + \frac{\beta_n^2}{\overline{\beta}_n \alpha_n}\boxed{\mathrm{E}_{q(x_0|x_n)}[(\epsilon_n - \hat{\epsilon}_n(x_n))^2]}.$$

Noise prediction residual
(NPR)

Generally, the mean $\mu_n(x_n) = \frac{1}{\sqrt{\alpha_n}}\left(x_n - \beta_n \frac{1}{\sqrt{\overline{\beta}_n}}\hat{\epsilon}_n(x_n)\right)$ is not optimal due to approximation or optimization error of $\hat{\epsilon}_n(x_n)$.

- Implementation framework of predicting NPR

最优协方差表达式: $\tilde{\boldsymbol{\sigma}}_n^*(\boldsymbol{x}_n)^2 = \lambda_n^2 \mathbf{1} + \gamma_n^2 \frac{\overline{\beta}_n}{\overline{\alpha}_n} \boxed{\mathbb{E}_{q(\boldsymbol{x}_0|\boldsymbol{x}_n)}[(\boldsymbol{\epsilon}_n - \hat{\boldsymbol{\epsilon}}_n(\boldsymbol{x}_n))^2]},$



基于预测噪声残差的最优协方差估计: $\hat{\boldsymbol{\sigma}}_n(\boldsymbol{x}_n)^2 = \lambda_n^2 \mathbf{1} + \gamma_n^2 \frac{\overline{\beta}_n}{\overline{\alpha}_n} \boxed{\boldsymbol{g}_n(\boldsymbol{x}_n)}.$

Song et al. Score-based generative modeling through stochastic differential equations, ICLR 2021.

- The continuous timesteps version (SDE)

- $q(x_0, \ldots, x_N)$ becomes
- $d\boldsymbol{x} = f(t)\boldsymbol{x}dt + g(t)d\boldsymbol{w} \leftrightarrow d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2\, \nabla \log q_t(\boldsymbol{x}))dt + g(t)d\overline{\boldsymbol{w}}$

- $p(x_0, \ldots, x_N)$ becomes
- $d\boldsymbol{x} = \left(f(t)\boldsymbol{x} - g(t)^2 \boldsymbol{s}_t(\boldsymbol{x})\right)dt + g(t)d\overline{\boldsymbol{w}}$

Conditional DPMs: Paired Data

We have pairs of (x_0, c), where x_0 is the data and c is the condition.
The goal is to learn the unknown conditional data distribution $q(x_0|c)$.

Conditional Model

- Original model $s_n(x_n) \to$ conditional model $s_n(x_n|c)$

- Training: $\min_{s_n} \mathrm{E}_c \mathrm{E}_n \bar{\beta}_n \mathrm{E}_{q_n(x_n|c)} \| s_n(x_n|c) - \nabla \log q_n(x_n|c) \|^2$

- Conditional DPM:
  - Discrete time: $p(x_{n-1}|x_n, c) = N(\mu_n(x_n|c), \Sigma_n(x_n))$, $\mu_n(x_n) = \frac{1}{\sqrt{\alpha_n}}\big(x_n + \beta_n s_n(x_n|c)\big)$
  - Continuous time: $d\boldsymbol{x} = \big(f(t)\boldsymbol{x} - g(t)^2 \boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c})\big)dt + g(t)d\bar{\boldsymbol{w}}$

- **Challenge**: design the model architecture $s_n(x_n|c)$

Discriminative Guidance

- Exact reverse SDE: $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2\,\nabla \log q_t(\boldsymbol{x}|c))dt + g(t)d\overline{\boldsymbol{w}}$

- $\nabla \log q_t(\boldsymbol{x}|c) = \underbrace{\nabla \log q_t(x)}_{\text{Original DPM}} + \underbrace{\nabla \log q_t(c|x)}_{\text{Discriminative model}}$

  Approximated by

  The paired data is used in the training of the discriminative model

- Conditional score-based SDE:
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(s_t(x) + \nabla \log p_t(c|x)))dt + g(t)d\overline{\boldsymbol{w}}$

- **Benefits**: Many discriminative models have well studied architectures

Scale Discriminative Guidance

- Exact reverse SDE: $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\nabla \log q_t(x) + \nabla \log q_t(c|x)))dt + g(t)d\overline{\boldsymbol{w}}$

- Scale discriminative guidance:
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\nabla \log q_t(x) + \lambda\nabla \log q_t(c|x)))dt + g(t)d\overline{\boldsymbol{w}}$

- Conditional score-based SDE:
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(s_t(x) + \lambda\nabla \log p_t(c|x)))dt + g(t)d\overline{\boldsymbol{w}}$
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(s_t(x|c) + \lambda\nabla \log p_t(c|x)))dt + g(t)d\overline{\boldsymbol{w}}$

Conditioned on label

Conditional	Guidance	Scale	FID	sFID	IS	Precision	Recall
✗	✗		26.21	**6.35**	39.70	0.61	0.63
✗	✓	1.0	33.03	6.99	32.92	0.56	**0.65**
✗	✓	10.0	**12.00**	10.40	**95.41**	**0.76**	0.44
✓	✗		10.94	6.02	100.98	0.69	**0.63**
✓	✓	1.0	**4.59**	**5.25**	186.70	0.82	0.52
✓	✓	10.0	9.11	10.93	**283.92**	**0.88**	0.32

Table 4: Effect of classifier guidance on sample quality. Both conditional and unconditional models were trained for 2M iterations on ImageNet 256×256 with batch size 256.

Dhariwal et al. Diffusion Models Beat GANs on Image Synthesis

Self Guidance

Ho et al. Unconditional Diffusion Guidance

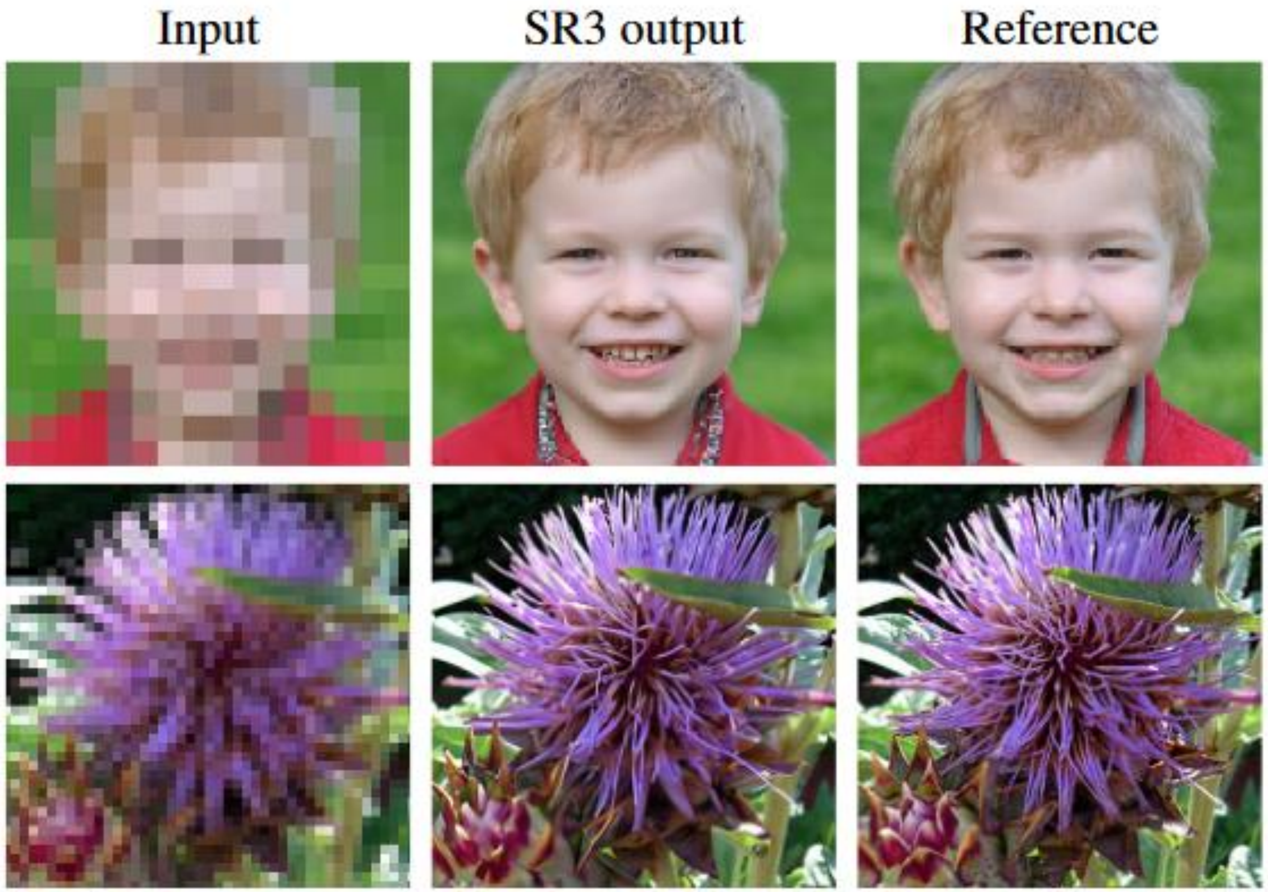
- Scale discriminative guidance: $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\nabla \log q_t(x) + \lambda \underbrace{\nabla \log q_t(c|x)}_{\text{Require an extra discriminative model}}))dt + g(t)d\overline{\boldsymbol{w}}$
- $\nabla \log q_t(c|x) = \nabla \log q_t(x|c) - \nabla \log q_t(x)$
- Learn conditional & unconditional model together
- Introduce token $\emptyset$, and use $s_t(x_t|\emptyset)$ to represent unconditional cases
- Conditional score-based SDE:
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(s_t(x|\emptyset) + \lambda(s_t(x|c) - s_t(x|\emptyset)))dt + g(t)d\overline{\boldsymbol{w}}$
- Training:
- $\min_{s_n(\cdot)} \mathrm{E}_c \mathrm{E}_n \bar{\beta}_n \underbrace{\mathrm{E}_{q_n(x_n|c)} \|s_n(x_n|c) - \nabla \log q_n(x_n|c)\|^2}_{\text{conditional loss}} + \lambda \underbrace{\mathrm{E}_n \bar{\beta}_n \mathrm{E}_{q_n(x_n)} \|s_n(x_n|\emptyset) - \nabla \log q_n(x_n)\|^2}_{\text{unconditional loss}}$

Application: Image Super-Resolution

- Paired data (x_0, c), x_0 is high resolution image, c is low resolution image

- Learn a conditional model $s_n(x_n|c)$
- **Architecture**: $s_n(x_n|c) = \mathrm{UNet}(\mathrm{cat}(x_n, c'), n)$, c' is the bicubic interpolation of c

Application: Image Super-Resolution



Nichol et al. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models

Application: Text to Image

- Dataset contains pairs of (x_0, c), where x_0 is image and c is text
- Techniques: conditional model with self-guidance
- Challenge: design $s_t(x|c)$

GLIDE (CF Guid.)

"a green train is coming down the tracks" — "a group of skiers are preparing to ski down a mountain." — "a small kitchen with a low ceiling" — "a group of elephants walking in muddy water." — "a living area with a television and a table"

Application: Text to Image

- Architecture of $s_t(x|c)$: UNet + Transformer
- UNet encodes image x
- Transformer encodes text c and the embedding is injected to UNet
  - The **token embedding** is injected after group normalization in Res Block:

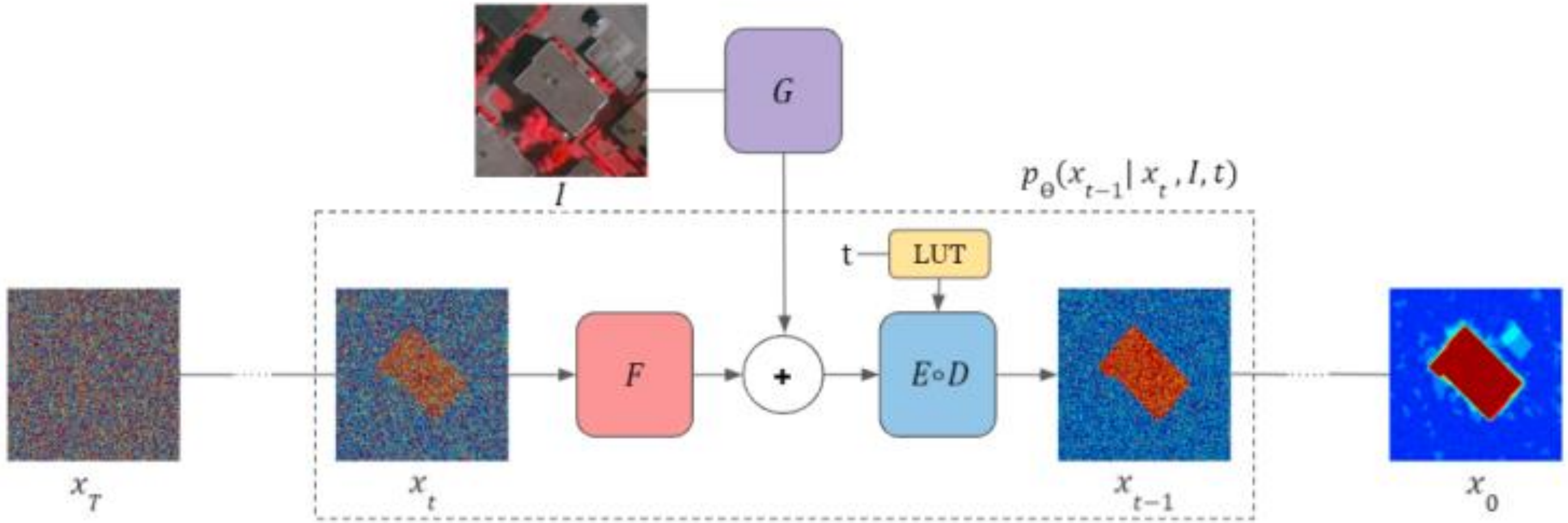
$$\mathrm{AdaGN}(h, y) = y_s \ \mathrm{GroupNorm}(h) + y_b$$

  - The **token embedding** is concatenated to the attention context in UNet

Other details
Dataset: the same as DALL-E
#parameters: 2.3 billion for 64x64

Application: Segmentation

- Paired data (x_0, c), x_0 is segmentation, c is image
- $s_t(x|c) = \mathrm{UNet}(F(x) + G(c), t)$



Conditional DPMs: Unpaired Data

We only have a set of x_0 (data).
The goal is to construct a conditional distribution $p(x_0|c)$.

Energy Guidance

- Unconditional DPM trained from a set of x_0 (data):
- $d\boldsymbol{x} = \left(f(t)\boldsymbol{x} - g(t)^2 \boldsymbol{s}_t(\boldsymbol{x})\right)dt + g(t)d\overline{\boldsymbol{w}}$

- A strategy to construct $p(x_0|c)$ is to insert an energy function:
- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\boldsymbol{s}_t(\boldsymbol{x}) - \nabla E_t(\boldsymbol{x}, c)))dt + g(t)d\overline{\boldsymbol{w}},\ x_T \sim p(x_T|c)$

- The generated data tends to have a low energy $E_t(\boldsymbol{x}, c)$
- The energy depends on specific applications

Energy Guidance

- Pros:
- Provides a framework for incorporating domain knowledge to DPMs

- Cons:
- $p(x_0|c)$ is very black box
- Energy design is based on intuition

Application: Text to Image

- High level idea: Define energy as a negative similarity between image and text

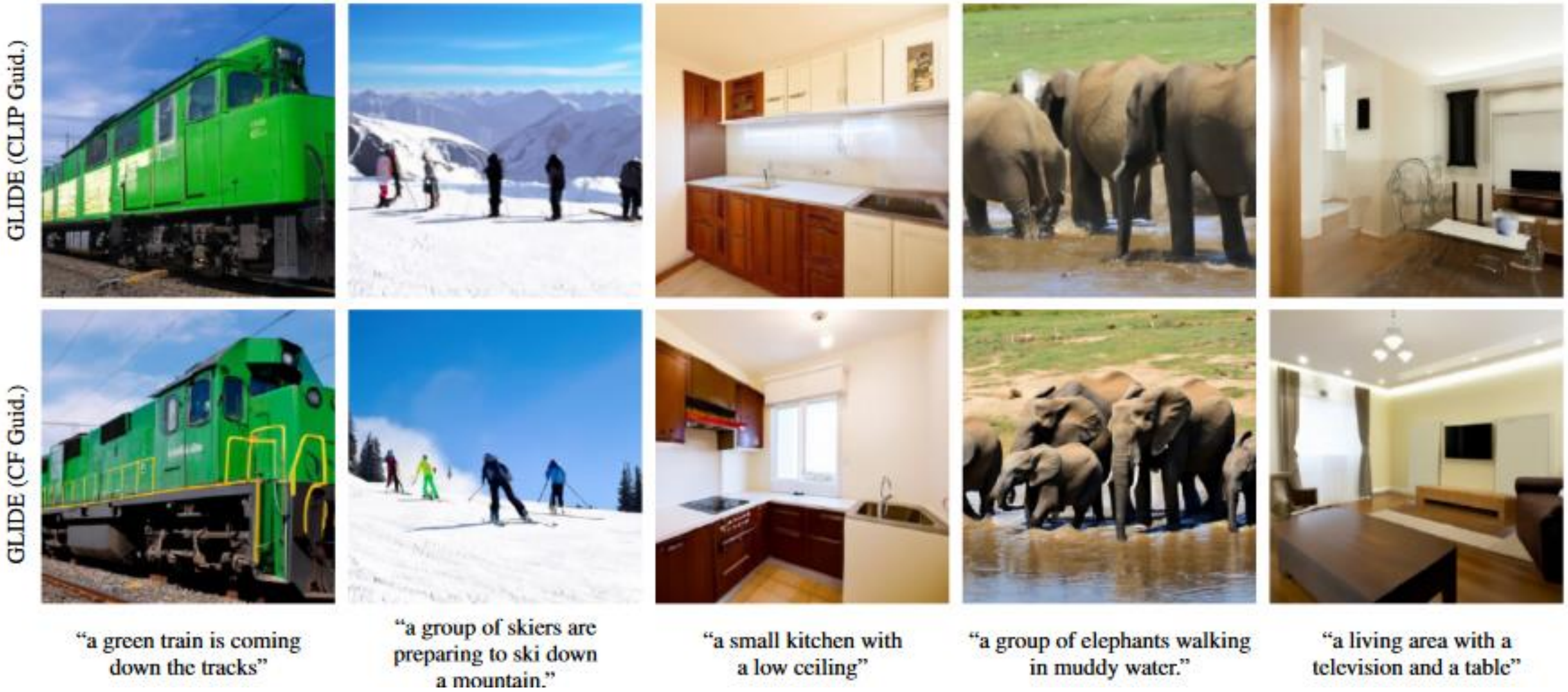
- CLIP provides a model to measure the similarity between images and texts:
- Similarity: $\mathrm{sim}(\boldsymbol{x}, c) = \boldsymbol{f}(\boldsymbol{x}) \cdot \boldsymbol{g}(c)$
- Energy: $E_t(\boldsymbol{x}, c) = -\mathrm{sim}(\boldsymbol{x}, c)$

Nichol et al. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models

Application: Text to Image

Energy guidance

Self guidance



Application: Generate Low Density Images

(i) High density

(ii) Low density

Dataset

Samples from SDE of $\boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c})$

Samples from SDE is more similar to high density part in dataset

Application: Generate Low Density Images

- Original SDE: $d\boldsymbol{x} = \left(f(t)\boldsymbol{x} - g(t)^2\boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c})\right)dt + g(t)d\overline{\boldsymbol{w}}$
- New SDE: $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c}) - \nabla E_t(\boldsymbol{x}, c)))dt + g(t)d\overline{\boldsymbol{w}}$

- High level intuition: Small energy ~ x is away from the class c
- $E_t(x, c) = \mathrm{sim}(x, c) = f(x) \cdot \mu_c$
  - f is an image encoder and μ_c is the averaged embedding of class c
  - Empirically, use a contrastive version of the loss

Vikash et al. Generating High Fidelity Data from Low-density Regions using Diffusion Models

Application: Generate Low Density Images

(i) High density

(ii) Low density

Dataset

Samples from SDE of $\boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c})$

Samples from $\boldsymbol{s}_t(\boldsymbol{x}|\mathrm{c}) - \nabla E_t(\boldsymbol{x}, c)$

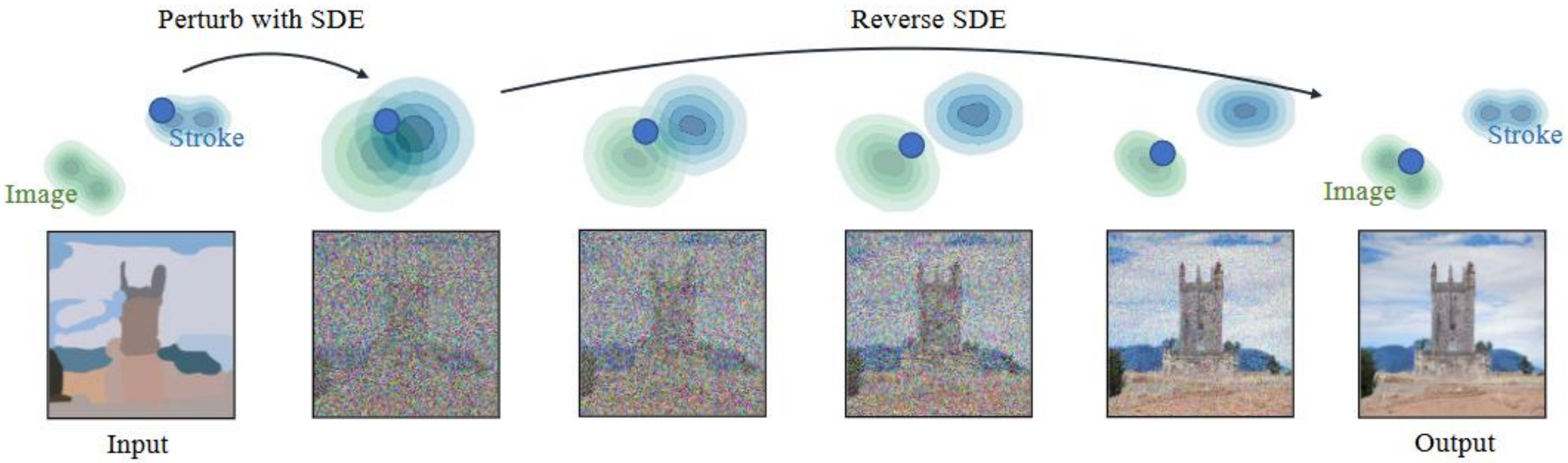
Application: Image2Image Translation

- c is the reference image
- $\boldsymbol{s}_t(\boldsymbol{x})$ is a DPM on target domain

- $d\boldsymbol{x} = (f(t)\boldsymbol{x} - g(t)^2(\boldsymbol{s}_t(\boldsymbol{x})))dt + g(t)d\overline{\boldsymbol{w}},\ x_{t_0} \sim p(x_{t_0}|c)$
  - No energy guidance
  - c only influence the start distribution
  - Choose an early start time $t_0 < T$
- $p(x_{t_0}|c)$ is a Gaussian perturbation of c

Application: Image2Image Translation

$p(x_{t_0}|c)$ is a Gaussian perturbation of c



Stroke to painting

DPMs for Downstream Tasks

Regard DPMs as pretrained models (feature extractors)

DPMs for Downstream Segmentation

DPM features are already unsupervised segmentation.

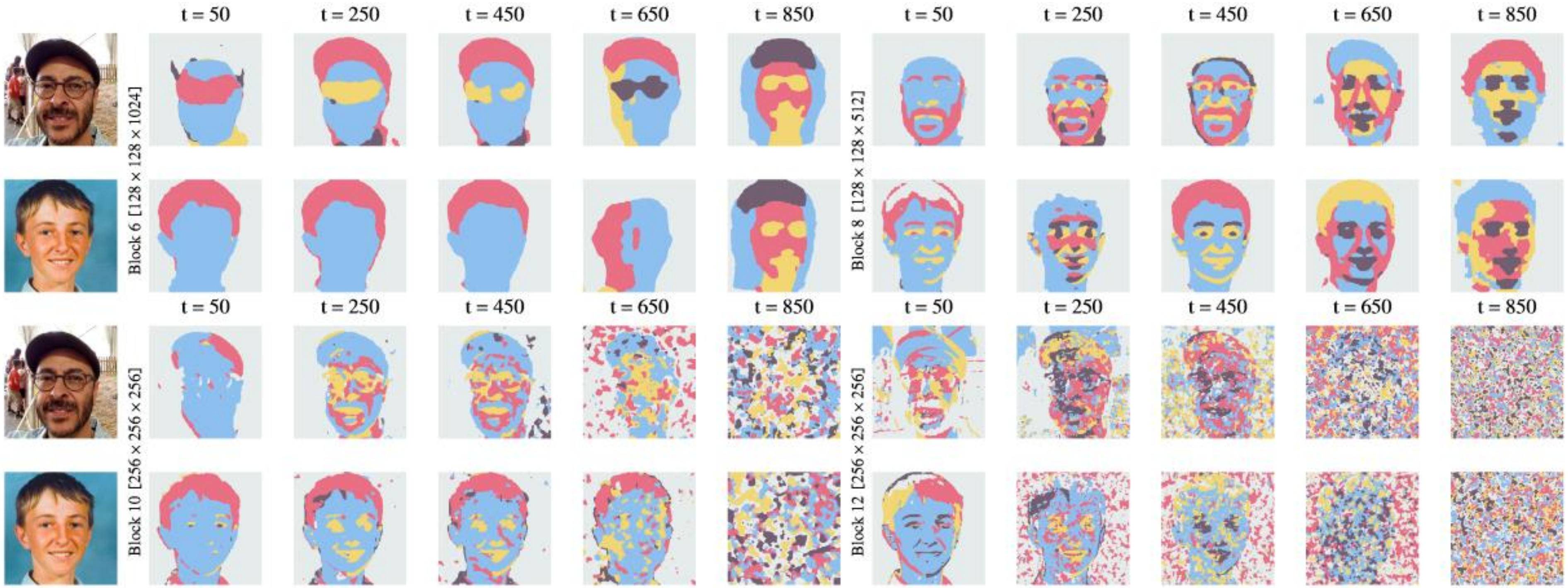
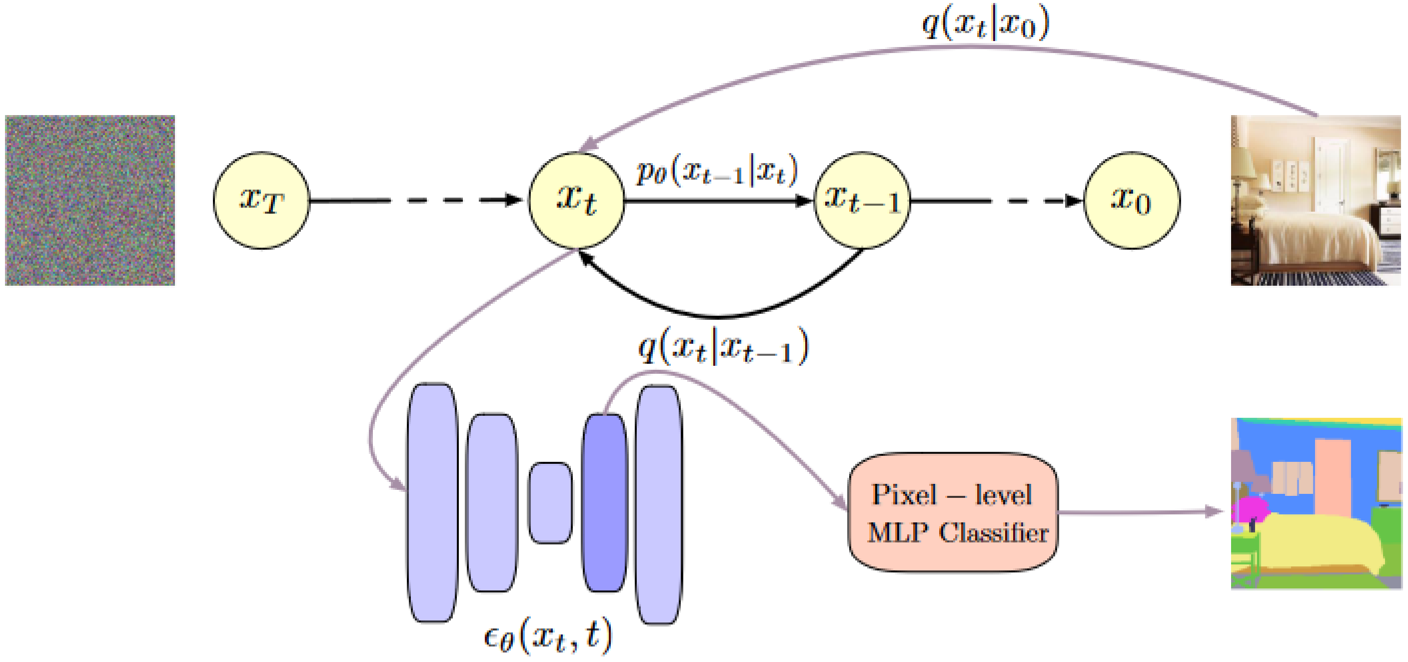


Figure 1: Examples of k-means clusters (k=5) formed by the features extracted from the UNet decoder blocks $\{6, 8, 10, 12\}$ on the diffusion steps $\{50, 250, 450, 650, 850\}$. The clusters from the middle blocks spatially span coherent semantic objects and parts.

DPMs for Downstream Segmentation

- Use features from DPMs at different layers and times.
- Finetune a MLP after these features.
- Only a small number of segmented data is required.



DPMs for Other Domains

DPMs for Other Domains

- Text to speech
  - *Vadim et. al. Grad-TTS: A Diffusion Probabilistic Model for Text-to-Speech*

Table 1: Unconditional generative modeling on UCF101 [36].

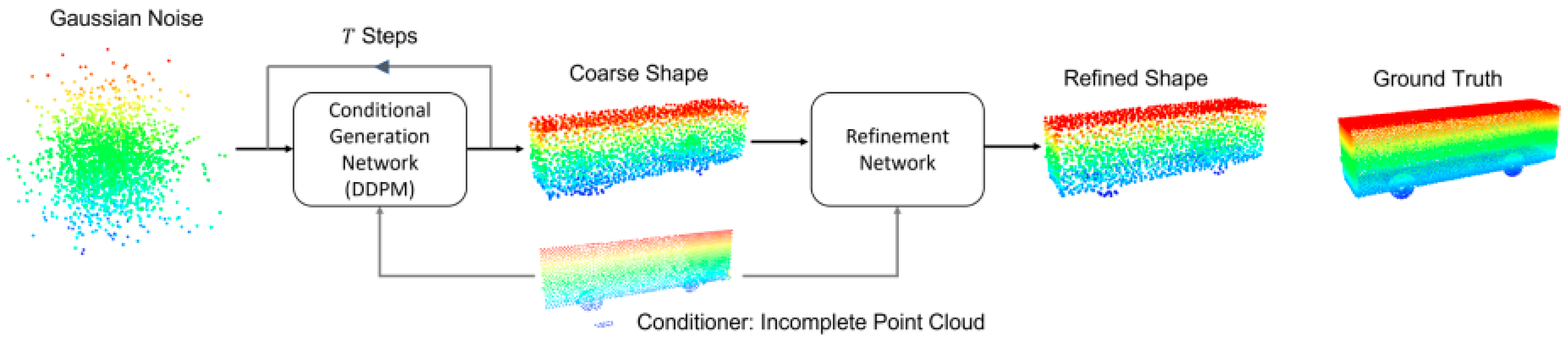
Method	resolution	FID↓	IS↑
MoCoGAN [38]	16x64x64	26998 ± 33	12.42
TGAN-F [17]	16x64x64	8942.63 ± 3.72	13.62
TGAN-ODE [12]	16x64x64	26512 ± 27	15.2
TGAN-F [17]	16x128x128	7817 ± 10	22.91 ± .19
VideoGPT [46]	16x128x128		24.69 ± 0.30
TGAN-v2 [28]	16x64x64	3431 ± 19	26.60 ± 0.47
TGAN-v2 [28]	16x128x128	3497 ± 26	28.87 ± 0.47
DVD-GAN [9]	16x128x128		32.97 ± 1.7
ours	16x64x64	**330**	**57.8 ± 1.3**
real data	16x64x64		59.4 ± 1.06

- Video generation
  - *Ho et. al. Video Diffusion Models*

- Industry anomaly detection
  - *Yana et.al. TFDPM: Attack detection for cyber-physical systems with diffusion probabilistic models* （网络物理系统的攻击检测）

DPMs for Other Domains

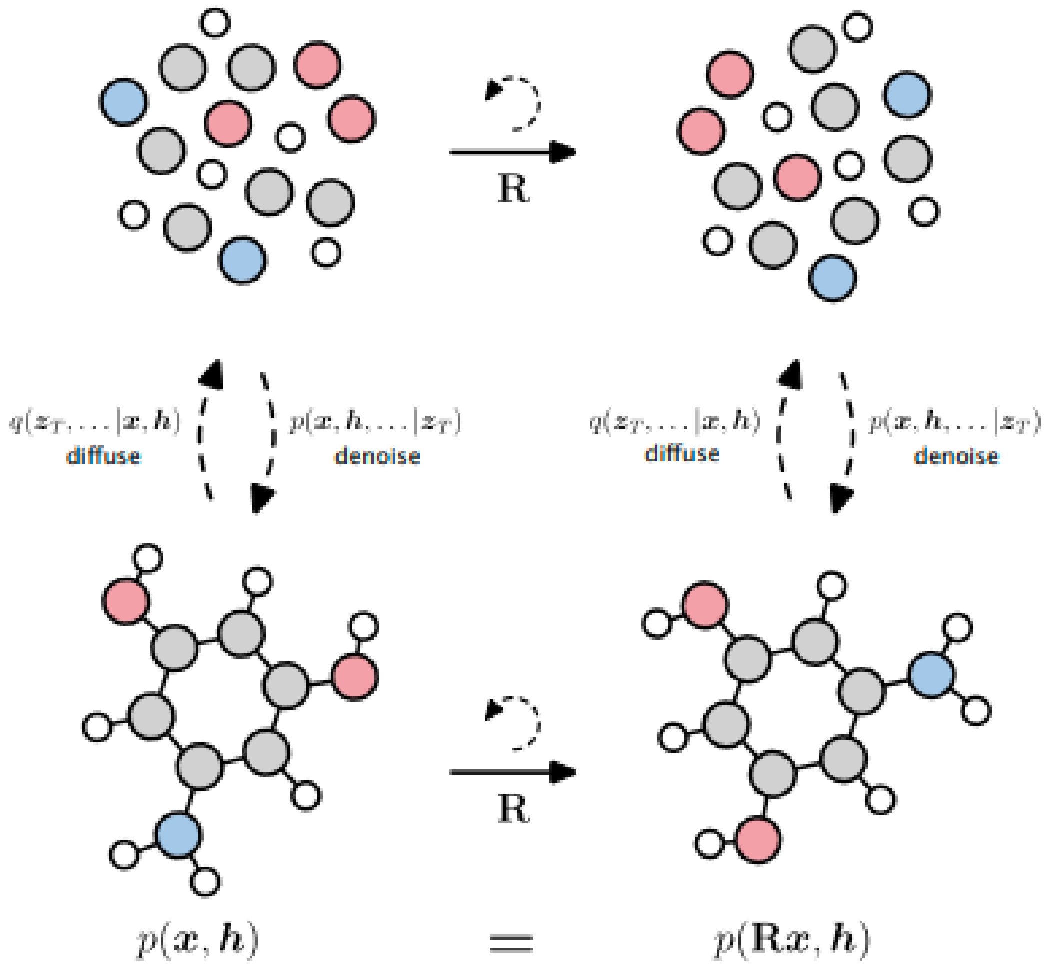
- Point Cloud
  - *Lyu et. al. A CONDITIONAL POINT DIFFUSION-REFINEMENT PARADIGM FOR 3D POINT CLOUD COMPLETION*



DPMs for Science

- Molecular dynamics
  - *Wang et.al. From data to noise to data: mixing physics across temperatures with generative artificial intelligence*
  - *Hoogeboom et. al. Equivariant Diffusion for Molecule Generation in 3D*

DPMs for Science



DPMs for Science

- Medical
  - *Aviles-Rivero et. al. Multi-Modal Hypergraph Diffusion Network with Dual Prior for Alzheimer Classification*

Thanks!