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

- Fast Sampling
- Mode Coverage/Diversity
- High Quality Samples
- Denoising Diffusion Models
- Likelihood-based models (Variational Autoencoders & Normalizing flows)

Generative Adversarial Networks (GANs)

Often requires 1000s of network evaluations!
WHAT MAKES A GOOD GENERATIVE MODEL?

The generative learning trilemma

Tackle the trilemma by accelerating diffusion models

- Fast Sampling
- High Quality Samples
- Mode Coverage/Diversity
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:

\[
q(x_{1:T} | x_0) = \prod_{t \geq 1} q(x_t | x_{t-1}), \quad q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t I)
\]

Forward diffusion process (fixed)

Data → Noise

Reverse denoising process (generative)

\[
p_\theta(x_{0:T}) = p(x_T) \prod_{t \geq 1} p_\theta(x_{t-1} | x_t), \quad p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma^2_t I)
\]

Trainable network (U-net, Denoising Autoencoder)

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.)!
TRAINING OBJECTIVE
Score matching for the cross entropy

\[ \mathcal{L}(x, \phi, \theta, \psi) = \mathbb{E}_{q_{\phi}(z_0|x)} \left[ -\log p_{\psi}(x|z_0) \right] + \text{KL}(q_{\phi}(z_0|x)||p_{\theta}(z_0)) \]

\[ = \mathbb{E}_{q_{\phi}(z_0|x)} \left[ -\log p_{\psi}(x|z_0) \right] + \mathbb{E}_{q_{\phi}(z_0|x)} \left[ \log q_{\phi}(z_0|x) \right] + \mathbb{E}_{q_{\phi}(z_0|x)} \left[ -\log p_{\theta}(z_0) \right] \]

- reconstruction term
- negative encoder entropy
- cross entropy

\[ CE(q(z_0|x)||p(z_0)) = \mathbb{E}_{t \sim \mathcal{U}[0,1]} \left[ \frac{g(t)^2}{2} \mathbb{E}_{q(z_t, z_0|x)} \left[ \| \nabla_{z_t} \log q(z_t|z_0) - \nabla_{z_t} \log p(z_t) \|_2^2 \right] \right] + \frac{D}{2} \log \left( 2\pi e \sigma_0^2 \right) \]

- time sampling
- Forward diffusion
- Diffusion kernel
- Trainable score function
- Constant
NORMAL DISTRIBUTION ASSUMPTION

Inductive biases for training

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

\[
CE(q(z_0 \mid x) \mid p(z_0)) = \mathbb{E}_{\tilde{t} \sim \mathcal{U}[0,1]} \left[ \frac{g(t)^2}{2} \mathbb{E}_{q(z_t \mid z_0 \mid x)} \left[ \| \nabla_{z_t} \log q(z_t \mid z_0) - \nabla_{z_t} \log p(z_t) \|_2^2 \right] \right]
\]

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

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/
Denoising Diffusion GANs

Zhisheng Xiao, Karsten Kreis, Arash Vahdat
ICLR 2021 (spotlight)
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

Diffused Data Distribution

True Denoising Distribution

$q(x_0)$ $q(x_1)$ $q(x_2)$ $q(x_3)$ $q(x_4)$ $q(x_5)$

$x_2 = X$

$q(x_t | x_5 = X)$
NORMAL ASSUMPTION IN DENOISING DISTRIBUTION

Holds for small steps

Denoising Process with Unimodal Normal Distribution

Denoising Process with Our Multimodal Conditional GAN

Data

Noise

Data

Noise
How can we parameterize the conditional GAN generator?

**Option 1**

**Option 2**

- Data
- Noise

- Option 1
- Option 2
ADVERSARIAL TRAINING

- How can we train the conditional GAN generator:

$$\min_{\theta} \sum_{t \geq 1} \mathbb{E}_{q(x_t)} [D_{\text{adv}}(q(x_{t-1} | x_t) || p_\theta(x_{t-1} | x_t))]$$
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.

\[
\min_{\theta} \sum_{t \geq 1} \mathbb{E}_{q(x_t)} \left[ D_{\text{adv}} \left( q(x_{t-1} | x_t) \ || \ p_\theta(x_{t-1} | x_t) \right) \right]
\]
EXPERIMENTAL RESULTS

CIFAR-10 dataset

Sample Quality vs. Sampling Time

CIFAR-10 Samples
RESULTS ON THE GENERATIVE LEARNING TRILEMMA

CIFAR-10 dataset

- DDPM
- DDIM
- StyleGAN ADA
- Ours

Quality (FID)

Sampling Time

1-Recall
OTHER DATASETS

CelebA-HQ 256

LSUN Churches Outdoor 256
SAMPLING TIME
CelebA-HQ 256

<table>
<thead>
<tr>
<th>Model</th>
<th>FID ↓</th>
<th># Fun. Calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Song et al. ICLR 2021</td>
<td>7.22</td>
<td>4000</td>
</tr>
<tr>
<td>Latent Space Diffusion Models, NeurIPS 2021</td>
<td>7.23</td>
<td>23</td>
</tr>
<tr>
<td>Denoising Diffusion GANs, ICLR 2022</td>
<td>7.60</td>
<td>2</td>
</tr>
</tbody>
</table>
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

Karsten Kreis’s Talk on Feb 17
THANK YOU!