DiffuseDRAW: Structured Latent Variables Model with Discrete Diffusion Prior

Jiahe Liu
Department of Electrical & Computer Engineering
The University of British Columbia
Vancouver, BC V6T1Z4
jiaheliu@ece.ubc.ca

Abstract

In contrast to conventional black-box image generative models like generative adversarial networks (GANs), structured models like Deep Recurrent Attention Writer (DRAW) and NP-DRAW aim at mimicking how humans draw in a part-by-part fashion, thus being more interpretable and facilitating more controllable generation. Inspired by the recent success of denoising diffusion probabilistic models(DDPMs) in black-box image generation, we explored DDPMs in improving the sample quality of structured generative models. We propose the DiffuseDRAW model, a structured image generative model with diffusion prior. To achieve this, we adapt the diffusion model to the discrete structured latent space and trained it with a pre-trained CNN encoder and decider. We evaluated the sample quality of the DiffuseDRAW on CIFAR-10 and CelebA datasets.

1 Introduction

The human perception of images is structured. When drawing a picture, what a human usually do is to draw a rough outline, then complete the painting part by part, and finally iteratively refine the painting. However, most existing image generative models like variational autoencoder(VAEs) and generative adversarial networks(GANs) generate an entire image at once, which we thought is unnatural. Therefore, a group of generative models, called the structured latent variable model, has been proposed to mimic how humans draw in a part-by-part fashion[1, 2, 3]. The discrete structured latent variables that decide whether to draw, what to draw and where to draw at each timestep make these models more interpretable and controllable. While the sample quality of structured models is limited by the expression of the discrete structured latent variables.

Recent work on denoising diffusion probabilistic models(DDPMs) exhibited high-quality image synthesis results and potential for the image generation task[4, 5]. Diffusion models defined a parameterized Markov chain to add random noise to data gradually through forward process and to construct desired data samples from the noise via reverse process. Meanwhile, due to the thousands of iterations needed to sample one single image, the sample speed of the diffusion models is slow. Song et al. 2022 shows it takes around 20 hours to sample 50k images of size 32×32 from a DDPM, but less than a minute to do so from a GAN on an Nvidia 2080 Ti GPU[6].

Inspired by the recent success of diffusion model, we proposed the DiffuseDRAW model that improves the structured latent variables model with the diffusion prior. The DiffuseDRAW uses the VAE framework and adapts the diffusion model to the discrete structured latent space to model the prior distribution. The diffusion model is applied to the latent space instead of being directly applied to the data space, which significantly accelerates the sampling process.

Preprint. Under review.

2 Related Work

Deep image generative models have been learned for decades and show the potential for learning complex empirical distributions of images. They can be roughly divided into two groups, explicit log-likelihood models like variational autoencoders(VAEs)[7, 8], normalizing flows(NFs)[9, 10, 11], autoregressive model[8, 12, 13], and deep diffusion models[4, 5]; implicit log-likelihood models like generative adversarial networks (GANs)[14, 15, 16]. Our work adapts the VAE framework while uses the structured latent variable refined with a diffusion model.

Structured latent variable model Structured latent variable models mimic how humans draw in a part-by-part way. The first paper proposed the structured latent variable model is Deep Recurrent Attentive Writer (DRAW)[1]. This model leverages the recurrent neural network as the encoder and decoder in VAE framework. The spatial attention mechanism is used in the encoder and decoder to decide where to read, where to write, and what to write. Attend-Inder-Repeat(AIR) adds the mechanism that allows the model to decide the appropriate inference and generation steps[2]. NP-DRAW model proposes a non-parametric prior distribution over the appearance of image parts so that the latent variable what-to-draw becomes a categorical random variable, which improves the expressiveness[3]. Besides, NP-DRAW uses a pre-trained transformer to model the prior.

Diffusion probabilistic model Diffusion models are inspired by non-equilibrium thermodynamics. They define a Markov chain and transitions of this chain are learned to reverse a diffusion process, which is a Markov chain that gradually adds noise to the data in the opposite direction of sampling until signal is destroyed. Diffusion models were first presented in the diffusion probabilistic model(DPMs)[4]. Then denoising diffusion probabilistic model(DDPM) proposed a novel noise-prediction reverse process parameterization and showed they are capable of generating high-quality samples[5]. In addition, the denoising score-matching method was proved to be equivalent to the diffusion probabilistic model when the time steps become infinite[17, 18].

Latent diffusion model To speed up the sampling process of diffusion models, several recent works explore applying them in the latent space. The Latent Score-based Generative Model (LSGM) proposed a novel approach to train a score-based generative model in the latent space[19]. Latent diffusion models (LDMs) applied the diffusion models in the latent space of VAE and introduced the cross-attention layers into the model architecture[20]. In addition, diffusion models are also leveraged in the discrete latent space. ImageBART[21] and Vector Quantised Discrete Diffusion Model (VQ-DDM)[22] both used the diffusion model to model the discrete prior in VQ-VAE.

3 Background

3.1 Structured latent variable model

NP-DRAW follows the general framework of VAEs. Instead of using continuous latent variables like VAE, NP-DRAW uses discrete sequential latent variables z, called non-parametric structured prior. Each z_t corresponds to the drawing of a part of the image at timestep t. In particular, at the t-th generation step, the group z_t describes an image part in terms of its location z_{loc}^t, its appearance z_{id}^t, and whether we draw it z_{is}^t on the latent canvas c_t.

What to draw For z_{id}, raw image patches with size $K \times K$ are first collected from the training dataset. Then a patch bank is built by applying the K-medoids clustering on those patches. So z_{id}^t indicates the index of a patch in the bank and is a categorical random variable.

Where to draw For simplicity, the image is discretized into a 2D grid so that a part can only center on a grid. Therefore, z_{loc}^t is also a categorical random variable.

Where to draw At each time step, the NP-DRAW model is allowed to choose whether to draw it on the canvas by sampling a per-step Bernoulli random variable z_{is}^t.

The NP-DRAW uses a Vision Transformer[23] to model the discrete structured prior. For the generation process, it samples from the prior and gets a sequential latent variable z. Then it applies a create-canvas function and gets a canvas that contains a rough outline of the image. Finally, the canvas is fed to the decoder to get the final image. Given an image x, it infers the latent variables z by an encoder during training. The model architecture is shown in Figure 1.

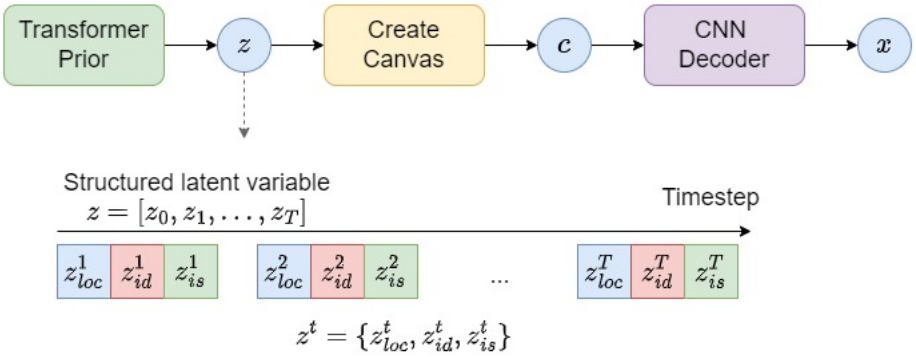


Figure 1: NP-DRAW model architecture

3.2 Denoising Diffusion Probabilistic Models

Diffusion models are latent variable generative models characterized by a forward and a reverse Markov process. Given a data point sampled from a real data distribution $x_0 \sim q(x)$, the forward diffusion process is defined as a fixed Markov chain, in which a small amount of Gaussian noise is added to the sample in T steps, producing a sequence of noisy samples $x_1, \ldots, x_T$. The noise level added in each step is controlled by a variance schedule $\beta_1, \ldots, \beta_T$. The data sample x_0 gradually loses its distinguishable features as step t becomes larger. Eventually, x_T is equivalent to pure Gaussian noise.

$$q(z_{1:T}|x_0) := \prod_{t=1}^{T} q(x_t|x_{t-1}), \quad q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}) \tag{1}$$

If it can reverse the above process and sample from the posterior $q(x_{t-1}|x_t)$, it will be able to recreate the true sample from a Gaussian noise input. Unfortunately, $q(x_{t-1}|x_t)$ is intractable. Therefore, the diffusion model learns a model p_θ to approximate these conditional probabilities. The reverse process is defined as a first-order Markov chain with a learned Gaussian transition distribution as follows.

$$p(x_{0:T}) := p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \quad p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{2}$$

Note that if time step T is large and β_t is small enough, $q(x_{t-1}|x_t)$ will approximate an isotropic Gaussian. The entire diffusion model can be trained end-to-end using variational inference.

4 Model

Our model follows the architecture of the NP-DRAW model, as shown in Figure 2. For the generation process, we sample a sequential latent variable z from the diffusion prior $p_\theta(z)$ and then apply a create-canvas function and get a canvas c that contains a rough outline of the image. Finally, we feed the canvas c to CNN decoder $p_\theta(x|z)$ generate image x conditioned on z. Given image x, the inference of latent variables z is implemented via CNN encoder $q_\Phi(z|x)$. For latent variable z, we use the same element z_{id} used in the NPDRAW model to capture the appearances of images. Hence, the prior distribution is a categorical distribution.

4.1 Discrete Diffusion Prior

We adapted the denoising diffusion model to the discrete structured latent space. The discrete latent variable z_{id} is a categorical random variable with K categories, which is equal to the size of the patch bank.

Forward process With forward transition matrices $[Q_t]_{ij} = q(z_t = i|z_{t-1} = j)$, the forward process can be written as a product of categorical distributions specified in terms of the probabilities over the

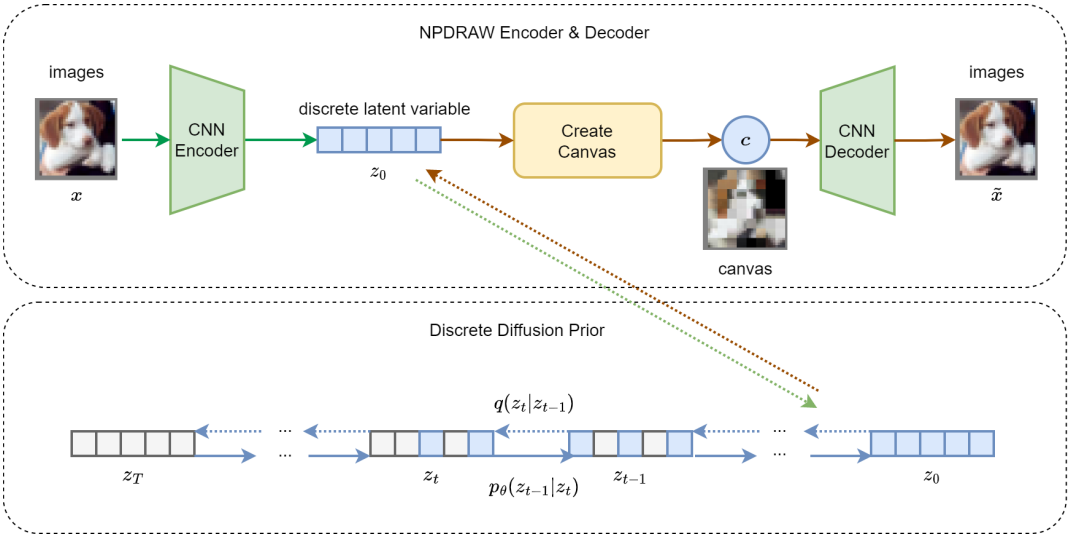


Figure 2: Overview over DiffuseDRAW model

patch bank indices:

$$q(z_{1:T}|z_0) = \prod_{t=1}^{T} q(z_t|z_{t-1}) = \prod_{t=1}^{T} Cat(z_t|p = z_{t-1}Q_t) \tag{3}$$

We follow transition matrices proposed in [24]:

$$[Q_t]_{ij} = (1-\beta_t)\mathbf{I} + \beta_t \mathbb{I}\mathbb{I}^T/K \tag{4}$$

$$= \begin{cases} 1 - \frac{K-1}{K}\beta_t & i = j \\ \frac{1}{K}\beta_t & i \neq j \end{cases} \tag{5}$$

where $\mathbb{I} = (1)_{j=1}^{K}$ is the all one vector. Since this transition matrix is doubly stochastic with strictly positive entries, the stationary distribution is uniform. Therefore, the transition probability of any entries to any other state is uniform at the end of the diffusion process.

As continuous DDPM, we can compute the marginal distribution of x_t at an arbitrary timestep t in a close form.

$$q(z_t|z_0) = Cat(z_t|p = z_0\overline{Q}_t),\ \overline{Q}_t = Q_1 Q_2 \dots Q_t \tag{6}$$

Note that the patch bank size K and timesteps T are not so large, $\overline{Q}_t$ can be simply precomputed for all t. Conditioned on x_0, the forward process posteriors are also tractable.

$$q(z_{t-1}|z_t, z_0) = \frac{q(z_t|z_{t-1}, z_0)q(z_{t_1}|z_0)}{q(z_t|z_0)} = cat(z_{t-1}|p = \frac{z_t Q_t^T \odot z_0 \overline{Q}_{t-1}}{z_0 \overline{Q}_t z_t^T}) \tag{7}$$

Reverse process We can sample the discrete latent variables z_0 starting from a uniform distribution $p(z_T) \sim U(0,1)$ via a reverse process of the diffusion Markov chain.

$$p_\theta = p(z_T) \prod_{t=1}^{T} p(z_{t-1}|z_t), \quad p_\theta(z_{t-1}|z_t) = Cat(z_{t-1}|p_\theta) \tag{8}$$

Instead of predicting the logits of $p_\theta(z_{t-1}|z_t)$ directly, we predict the logits of a distribution $\tilde{p}_\theta(\tilde{z}_0|z_t)$ using a neural network. Hence, the distribution $p_\theta(z_{t-1}|z_t)$ can be computed by the following parameterization.

$$p_\theta(z_{t-1}|z_t) \propto \sum_{\tilde{z}_0} q(z_{t_1}|z_t, \tilde{z}_0)\tilde{p}_\theta(\tilde{z}_0|z_t) \tag{9}$$

Table 1: Comparison of sample qualities (lower FID score is better).

Method	CIFAR-10 32×32	Celeba (μm) 64×64
VAE	106.70	70.00
2sVAE	72.90	44.4
NVAE	55.97	14.74
snGAN	14.20	-
WGAN	54.82	40.29
WGAN GP	42.18	30.30
DDPM	3.17	-
ImageBART	-	-
VQ-DDM	-	5.64
PixelCNN++	68.00	72.46
AIR	673.93	399.41
DRAW	162.00	157.00
NP-DRAW	62.72	41.87
DiffuseDRAW(ours)	69.72	46.05

4.2 Training Objective

Since it is not easy to train the VAE framework and diffusion prior jointly, we applied the two-stage training method. Firstly, we pre-trained the CNN encoder and decoder using the transformer prior in the NPDRAW by maximizing the loss function proposed in [3].

$$\mathcal{L}_{np} = \mathbb{E}_{q(z_0|x)}[\log(p(x|z_0))] - D_{KL}(q(z_0|x)||p_{np}(z_0)) - D_{KL}(p_h(z_0|x)||p(z_0|x)) \tag{10}$$

where $p_{np}(z_0)$ is the prior distribution in the NPDRAW model and $p_h(z_0|x)$ is the general probabilistic parsing of latent variables z given an image x via Heuristic Parsing Algorithm proposed in [3].

Then we sampled the latent variable z_0 through the pre-trained CNN encoder as ground truth to train the diffusion model. we typically optimize the variational upper bound on the negative log-likelihood.

$$\begin{aligned}\mathcal{L}_{vb} = \mathbb{E}_{q(z_0|x)}[D_{KL}[q(z_T|z_0)||p(z_T)] + \sum_{t=2}^{T} \mathbb{E}_{q(z_t|z_0)}[D_{KL}[q(z_{t-1}|z_t, z_0)||p_\theta(z_{t-1}|z_t)]] \\ -\mathbb{E}_{q(z_1|z_0)}[\log p_\theta(z_0|z_1)]]\end{aligned} \tag{11}$$

5 Experiments

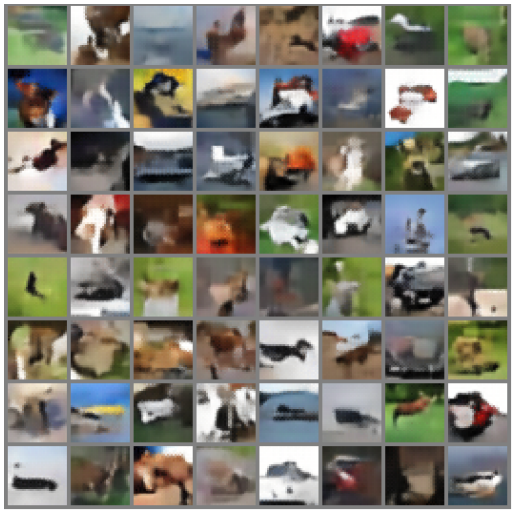
To train the CNN encoder and decoder, we follow the same training and evaluation setup as used in [3]. When training the diffusion prior, we set the patch bank size K as 200 and timesteps $T = 1000$ for all experiments. The variance schedule $\beta_1, \beta_2, \ldots, \beta_T$ is set to constants increasing linearly from $\beta_1 = 0.02$ to $\beta_T = 1.0$.

Since the latent variables are sequential, we use one-dimensional U-Net to predict logits of a distribution $\tilde{p}_\theta(\tilde{z}_0|z_t) = Cat(z_{t-1}|p_\theta)$. The model has four feature map resolutions and two one-dimensional convolutional residual blocks for each resolution level. The output of the 1D U-Net is $logits = nn_\theta(normalize(x_t^{int})) + x_t^{one-hot}$, where x_t^{int} and $x_t^{one-hot}$ denote integer and one-hot representations of x_t respectively.

Dataset We evaluate our model on the image dataset CIFAR-10 and CelebA. The CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes of common objectives, like birds, cats, and airplanes. While CelebA is a large human face image dataset.

Baselines We compare our model with three classes of generative models.

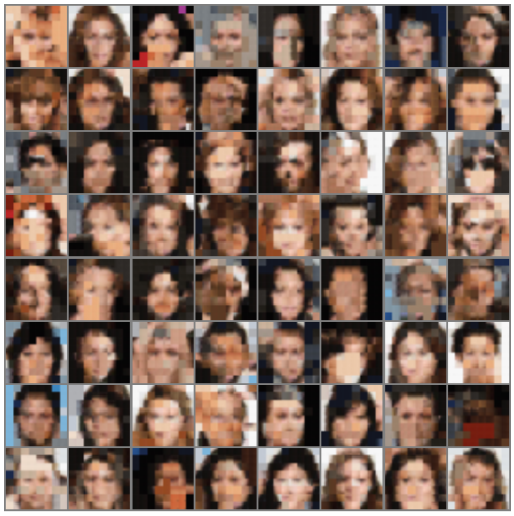
1. Structured generative model: DRAW[1], AIR[2], PixelCNN++[13], VQ-DRAW[25], and NP-DRAW[3].



(a) sampled images on CIFAR-10

(c) sampled images on CelebA

(b) sampled canvas on CIFAR-10



(d) sampled canvas on CelebA

Figure 3: Visualization of sampled canvases and images generated by DiffuseDRAW

2. Diffusion probabilistic model: DDPM[5], ImageBART[21], and VQ-DDM[22].
3. Generic generative model: VAE[7], 2sVAE[26], NVAE[8], WGAN[15], snGAN[27], WGAN-GP[16].

Evaluation Metric For all experiments, we compute the FID score to evaluate the quality and diversity of sampled images[28]. We draw 10K samples from each model and compute the FID score w.r.t. 10K images in the test set.

5.1 Image Generation

We compared the DiffuseDRAW model with all baselines on the unconditional image generation task in terms of FID score, as shown in Table 1. We also provide more visualization of our generated canvases and images in Figure 3.

As we can see from Table 1, our model outperforms most previous structured image generative models, except the NPDRAW. This indicates our model still needs to be improved. One factor that limits the sample quality of VAE architecture models is the prior-hole problem, i.e., the mismatch

between the VAE prior $p(z)$ and the aggregated posterior $q(z|x)$. With a powerful prior model like the diffusion models, if we could train the prior to approximate the posterior well enough, it should alleviate the prior-hole problem and improve the sample quality. Therefore, DiffuseDRAW could outperform the previous NPDRAW with proper improvements.

Note that there is a huge gap between the performances of the structured image generative models and denoising diffusion probabilistic models. The diffusion model indeed shows impressive high-quality image generation ability. However, the continuous latent variables in DDPMs lack interpretability. Hence, DDPMs may fail on some controllable image generation tasks, such as image editing/composition. While the structured latent variables are more intuitive and interpretable for humans, facilitating more controllable generation.

6 Conclusion

In this paper, we propose the DiffuseDRAW model, a structured image generative model with diffusion prior. We adapted the diffusion model to discrete structured latent space and propose a two-stage training method to train the VAE framework and diffusion prior. In the unconditional image generation experiments, our model is comparable with previous structured image generative models. However, it still needs to be improved.

For future work, we will combine the diffusion process with the transformer, which is more suitable for sequential data than U-Net. In addition, we will use all elements of the structured latent variables, including whether to draw, what to draw, and where to draw. To archive this, we should find a novel approach for diffusion prior to model the dependency of different elements. Moreover, we will explore a novel hybrid (discrete-continuous) latent variable to further improve the sample quality.

References

[1] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In *International conference on machine learning*, pages 1462–1471. PMLR, 2015.

[2] SM Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. Attend, infer, repeat: Fast scene understanding with generative models. *Advances in Neural Information Processing Systems*, 29, 2016.

[3] Xiaohui Zeng, Raquel Urtasun, Richard Zemel, Sanja Fidler, and Renjie Liao. Np-draw: A non-parametric structured latent variable model for image generation. In *Uncertainty in Artificial Intelligence*, pages 1089–1099. PMLR, 2021.

[4] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[6] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[7] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[8] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in Neural Information Processing Systems*, 33:19667–19679, 2020.

[9] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.

[10] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[11] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[12] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29, 2016.

[13] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[15] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[16] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

[17] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

[18] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[19] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

[20] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[21] Patrick Esser, Robin Rombach, Andreas Blattmann, and Bjorn Ommer. Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. *Advances in Neural Information Processing Systems*, 34:3518–3532, 2021.

[22] Minghui Hu, Yujie Wang, Tat-Jen Cham, Jianfei Yang, and Ponnuthurai N Suganthan. Global context with discrete diffusion in vector quantised modelling for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11502–11511, 2022.

[23] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[24] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

[25] Alex Nichol. Vq-draw: A sequential discrete vae. *arXiv preprint arXiv:2003.01599*, 2020.

[26] Bin Dai and David Wipf. Diagnosing and enhancing vae models. *arXiv preprint arXiv:1903.05789*, 2019.

[27] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[28] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.