J. Korean Soc. Ind. Appl. Math. Vol.28, No.1, 1–21, 2024 https://doi.org/10.12941/jksiam.2024.28.001

REVIEW OF DIFFUSION MODELS: THEORY AND APPLICATIONS

HYUNGJIN CHUNG[1], HYELIN NAM[2], AND JONG CHUL YE[2†]

[1]DEPARTMENT OF BIO & BRAIN ENGINEERING, KAIST, SOUTH KOREA
Email address: hj.chung@kaist.ac.kr

[2]KIM JAE CHUL GRADUATE SCHOOL OF AI, KAIST, SOUTH KOREA
Email address: {hyelin.nam, †jong.ye}@kaist.ac.kr

ABSTRACT. This review comprehensively explores the evolution, theoretical underpinnings, variations, and applications of diffusion models. Originating as a generative framework, diffusion models have rapidly ascended to the forefront of machine learning research, owing to their exceptional capability, stability, and versatility. We dissect the core principles driving diffusion processes, elucidating their mathematical foundations and the mechanisms by which they iteratively refine noise into structured data. We highlight pivotal advancements and the integration of auxiliary techniques that have significantly enhanced their efficiency and stability. Variants such as bridges that broaden the applicability of diffusion models to wider domains are introduced. We put special emphasis on the ability of diffusion models as a crucial foundation model, with modalities ranging from image, 3D assets, and video. The role of diffusion models as a general foundation model leads to its versatility in many of the downstream tasks such as solving inverse problems and image editing. Through this review, we aim to provide a thorough and accessible compendium for both newcomers and seasoned researchers in the field.

1. INTRODUCTION

Deep generative models [1, 2, 3, 4, 5, 6, 7] have recently undergone a transformative journey, emerging as pivotal tools for modeling the prior distribution of data explicitly or implicitly through a parametrized neural network, $p_{\rm data} \approx p_\theta(\boldsymbol{x})$. Among these generative models, diffusion models [5, 6, 7] have ascended to prominence, heralded for their exceptional capability to generate high-fidelity samples without mode collapse or adversarial training as in generative adversarial networks [1]. This paper embarks on a comprehensive review of diffusion models, delineating their theoretical foundations and recent applications.

Diffusion models generate data by reversing the forward Gaussian noising trajectory, which can be described by a forward stochastic differential equation (SDE). The reverse generative process is thus a process that generates data starting from pure Gaussian noise, akin to most other types of generative models, such as GAN [1], VAE [2], and Normalizing Flows [4]. One of the key differences of diffusion models against other generative models is that the

Received March 23 2024; Accepted March 25 2024; Published online March 25 2024.
2000 *Mathematics Subject Classification.* 60–02.
Key words and phrases. Diffusion models, Generative models.
† Corresponding author.

generative trajectory is predetermined as a denoising trajectory, governed by the Stein score function [8], parametrized through a neural network $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \approx \boldsymbol{s}_\theta(\boldsymbol{x}, t)$. To sample data from noise, one solves a differential equation (DE) numerically by iteratively querying the score function across the reverse time horizon. The process of iterative refinement, achieved through a parameter-shared neural network, enables stable and high-fidelity sampling as opposed to other one-step generative models, radically mitigating the limited expressivity of the parametrized neural networks.

Due to these advantages, diffusion models have risen as the de facto standard of generative models, especially in the vision community, where the applications span across image [9, 10, 11], 3D [12, 13], video [14, 15, 16], and even 4D [17]. Currently, diffusion models are to vision what autoregressive language models are to language: it is a local minimum so good that it almost seems hard to escape without a significant momentum in the whole research community. They are exceptionally robust and scalable, allowing them to be leveraged as vision foundation models, that can be used as a fully general prior that can be easily fine-tuned or leveraged as a plug-and-play module for downstream applications such as editing or inverse problem-solving.

This review aims to navigate the intricate landscape of diffusion models, offering insights into their theoretical mechanics, developmental milestones, and the cutting-edge applications that illustrate their transformative potential. The remainder of this paper is organized as follows: Sec. 2 delves into the theoretical underpinnings of diffusion models, elucidating the principles and processes that define their operation. Sec. 3 discusses various diffusion trajectories that can be modeled other than the standard Gaussian diffusion in the original signal space. Sec. 4 is devoted to different sampling methods that diffusion models can take, ranging from standard DE solvers to various distillation techniques. Sec. 5 summarize the emerging applications in the field, focusing on text-to-x foundation models and inverse problem solving. We conclude the review with future perspectives and its societal impact in Sec. 6.

2. THEORY

2.1. **Score perspective.** Consider the following continuous diffusion process $\boldsymbol{x}(t), t \in [0, T]$ with $\boldsymbol{x}(t) \in \mathbb{R}^d$ [5]. We set $\boldsymbol{x}(0) \sim p_0(\boldsymbol{x})$, where $p_0 = p_{\text{data}}$ as our initial data distribution, and $\boldsymbol{x}(T) \sim p_T$, where p_T is a reference distribution that we can sample from. The forward noising process from $t = 0 \to T$ can be defined by the following Itô stochastic differential equation:

$$d\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, t)dt + g(t)d\boldsymbol{w}, \quad \boldsymbol{f} : \mathbb{R}^d \times \mathbb{R} \mapsto \mathbb{R}^d,\ g : \mathbb{R} \mapsto \mathbb{R}, \tag{2.1}$$

where $\boldsymbol{f}$ is the drift function of $\boldsymbol{x}(t)$, g is the diffusion coefficient coupled with the standard d-dimensional Brownian motion $\boldsymbol{w} \in \mathbb{R}^d$. By properly choosing $\boldsymbol{f}, g$, one can asymptotically approach the Gaussian distribution as $t \to T$. When the drift function $\boldsymbol{f}$ is taken to be an affine function of $\boldsymbol{x}$, i.e. $\boldsymbol{f}(\boldsymbol{x}, t) = f(t)\boldsymbol{x}$, then the perturbation kernel $p(\boldsymbol{x}(t)|\boldsymbol{x}(0))$ is always Gaussian, where the parameters can be calculated in closed-form. Hence, perturbing the data with the perturbation kernel $p(\boldsymbol{x}(t)|\boldsymbol{x}(0))$ can be done without running the forward SDE. Owing to this property, one never *gradually* adds noise to data when training a diffusion model.

For given forward SDE in Eq. (2.1), it can be shown that there exists a reverse-time SDE running backwards [5, 18, 19]:

$$d\boldsymbol{x} = [\boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})]dt + g(t)d\bar{\boldsymbol{w}} \tag{2.2}$$

where dt is the infinitesimal *negative* time step, and $\bar{\boldsymbol{w}}$ is the standard Brownian motion running backwards. Running the reverse diffusion in Eq. (2.2) by sampling a random gaussian noise as an initial value would lead to sampling from $p_0(\boldsymbol{x})$. In order to do so, it is clear that we need access to the time-conditional score function $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$, which corresponds to the score function of the smoothed data distribution that is convolved with a Gaussian kernel.

An interesting fact is that there exists a corresponding deterministic ODE to Eq. (2.2), which reads

$$d\boldsymbol{x} = [\underbrace{\boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})}_{=:\tilde{\boldsymbol{f}}_\theta(\boldsymbol{x}, t)}]dt. \tag{2.3}$$

The ODE in Eq. (2.3) is called probability-flow ODE (PF-ODE). While Eq. (2.2) and Eq. (2.3) recover the same law $p_t(\boldsymbol{x})$, PF-ODE has several intriguing properties. First, diffusion models can now be seen as a type of continuous normalizing flows (CNF) [3], by considering the network as $\tilde{\boldsymbol{f}}_\theta$, leading to tractable likelihood computation. Second, ODE solvers are typically more well-behaved compared to SDE solvers. As will be discussed in further detail in Sec. 4.1, solving the PF-ODE instead of the reverse SDE leads to faster sampling.

One can train a neural network to approximate the actual score function via a procedure called score matching [20, 5] to estimate $\boldsymbol{s}_\theta(\boldsymbol{x}, t) \approx \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$, and plug it into Eq. (2.2). However, it is known that using explicit or implicit score matching is hardly scalable due to the instability and the compute requirements [20]. To circumvent technical difficulties, denoising score matching (DSM) is used

$$\theta^* = \arg\min_\theta \mathbb{E}_{t\sim\mathrm{Unif}(0,T), \boldsymbol{x}_t\sim p(\boldsymbol{x}_t|\boldsymbol{x}_0), \boldsymbol{x}_0\sim p(\boldsymbol{x}_0)} \left[\|\boldsymbol{s}_\theta(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{x}_0)\|_2^2\right]. \tag{2.4}$$

It should be noted that DSM, as the name implies, is equivalent to training a denoising autoencoder (DAE) on multiple noise levels, determined by an additional input t. Concretely, consider the simplest forward perturbation kernel $p(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{x}_0, t^2\boldsymbol{I})$[1]. Then, by setting a denoiser parametrization $D_\theta(\boldsymbol{x}_t, t) \triangleq -\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)/t^2$, it is easy to see that Eq. (2.4) can be rewritten as

$$\theta^* = \arg\min_\theta \mathbb{E}_{t\sim\mathrm{Unif}(0,T), \boldsymbol{x}_t\sim p(\boldsymbol{x}_t|\boldsymbol{x}_0), \boldsymbol{x}_0\sim p(\boldsymbol{x}_0)} \left[t\|D_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|_2^2\right]. \tag{2.5}$$

The equivalence between Eq. (2.4) and Eq. (2.5) is also related to Tweedie's theorem [21, 22]

Theorem 1 (Tweedie's theorem)**.** *Given a perturbation kernel $p(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{x}_0, t^2\boldsymbol{I})$, the posterior mean is given as*

$$\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \boldsymbol{x}_t + t^2 \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t)$$

[1]This choice is called the variance exploding (VE) diffusion, as the signal is kept the same throughout the diffusion process, but buried under exploding noise.

In other words, the parametrization in Eq. (2.5) is a way of directly estimating the posterior mean $\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$. Regardless of the parametrization and thanks to 1, diffusion models can be seen as having two dual representations: the noisy variable $\boldsymbol{x}_t$ that evolves with the reverse SDE in Eq. (2.2), and the posterior mean $\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$, which is implicitly given by the Tweedie's theorem, and can be thought of as the end point of the trajectory when taking a tangent direction to the current step.

2.2. **Variational perspective.** Parallel to the development of the score-based perspective on diffusion models, a variational perspective was also developed [6, 7], which now links diffusion models to VAEs [2]. Specifically, under this perspective, diffusion models are a hierarchical latent variable model called denoising diffusion probabilistic models (DDPM)

$$p_\theta(\boldsymbol{x}_0) = \int p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta^{(t)}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\, d\boldsymbol{x}_{1:T},$$

where $\boldsymbol{x}_{\{1,\dots,T\}} \in \mathbb{R}^d$. The neural network that models p_θ is then trained by minimizing the evidence lower bound (ELBO)

$$\mathbb{E}[-\log p_\theta(\boldsymbol{x}_0)] \le \mathbb{E}_q \left[-\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \right] = \mathbb{E}_q \left[-\log p(\boldsymbol{x}_T) - \sum_{t\ge 1} \log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})} \right] \tag{2.6}$$

where the inference distribution q is defined by the Markovian forward conditional densities

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t|\sqrt{\beta_t}\boldsymbol{x}_{t-1}, (1-\beta_t)I), \tag{2.7}$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)I).$$

Here, the noise schedule β_t is a strictly monotonically increasing sequence of t, with $\bar{\alpha}_t := \prod_{i=1}^{t} \alpha_t$, $\alpha_t := 1 - \beta_t$. The noise schedule is chosen such that the signal coefficient $\sqrt{\bar{\alpha}_t}$ is sufficiently close to 0 as $t \to T$, which in turn ensures that the noise coefficient $1 - \bar{\alpha}_t$ is sufficiently close to 1, approaching the standard normal distribution. Unlike the choice of VE diffusion discussed in Sec. 2.1, the choice made here is called variance preserving (VP). Interestingly, the discrete VP setup in Eq. (2.7), when pushed to the continuous counterpart by setting the number of discretization steps to $N \to \infty$, leads to the following SDE

$$d\boldsymbol{x} = -\frac{1}{2}\beta_t \boldsymbol{x}\, dt + \sqrt{\beta_t} d\boldsymbol{w}. \tag{2.8}$$

Minimizing the ELBO objective in Eq. (2.6) essentially leads to the following optimization problem

$$\min_\theta \mathbb{E}_q \left[\sum_{t>1} D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)||p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)) \right]. \tag{2.9}$$

The KL minimization problem in Eq. (2.9) is tractable as both distributions are Gaussians. For the first term, this comes from Bayes rule and the Markov property

$$q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0)\frac{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}{q(\boldsymbol{x}_t|\boldsymbol{x}_0)} = \mathcal{N}(\boldsymbol{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \boldsymbol{x}_0), \tilde{\beta}_t \boldsymbol{I}),$$

$$\text{where} \quad \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \boldsymbol{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t, \; \tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t.$$

For the second term, the reverse distribution is Gaussian as we are considering small perturbations for a single step of forward diffusion [7]. A typical parametrization is to set

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \tilde{\beta}\boldsymbol{I}),$$

$$\text{where} \quad \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right).$$

Under this choice, the ELBO objective in Eq. (2.6) can be simplified to the epsilon-matching objective by ignoring the time-dependent weighting factors

$$\theta^* = \arg\min_\theta \mathbb{E}_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0), \boldsymbol{x}_0 \sim p_{\text{data}}(\boldsymbol{x}_0), \boldsymbol{\epsilon}\sim\mathcal{N}(0,\boldsymbol{I})}\left[\|\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{\epsilon}\|_2^2\right]. \tag{2.10}$$

Epsilon matching is equivalent to DSM/DAE objective up to a constant with different parametrization. Specifically, in the VP-case, $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) = -\sqrt{1-\bar{\alpha}_t}\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$. Given the equivalence of the forward noising distribution in Eq. (2.8) and the learning objective in Eq. (2.4),Eq. (2.10), it can be seen that the two perspectives essentially lead to the same model.

Inference can be done by plugging in the trained $\boldsymbol{\epsilon}_\theta$ to estimate the mean of $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$, leading to the following iteration

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right) + \tilde{\beta}_t\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}).$$

Notice that similar to the reverse SDE in Eq. (2.2), we add stochastic noise in every iteration during DDPM sampling, leading to slower inference. A canonical way to avoid this, similar to the transition to the PF-ODE, can be done by denoising diffusion implicit models (DDIM) [5], where another inference distribution is introduced

$$q_\eta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\eta\tilde{\beta}_t^2}\frac{\boldsymbol{x}_t - \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0}{\sqrt{1-\bar{\alpha}_t}}, \eta\tilde{\beta}_t^2\boldsymbol{I}),$$

where $\eta \in [0, 1]$. By setting $\eta = 1.0$, we recover the original DDPM sampling with maximal stochasticity. By setting $\eta = 0.0$, we achieve a deterministic sampler, which can be shown to be equivalent to the VE PF-ODE [5]. Using smaller values of η leads to better results when the aim is to reduce the number of function evaluations (NFE).

3. Diffusion Trajectories

3.1. **Latent Diffusion.** Diffusion models are compute-heavy. This is not only because diffusion models require sequentially querying diffusion models to numerically solve the generative SDE/ODE, but also because the *latent* $\boldsymbol{x}_t$ has the same dimension as the original signal $\boldsymbol{x}_0$.

This makes directly scaling diffusion models to high-dimensional signals hard, requiring special treatments to achieve decent results [23, 24]. Also, this is different from most other generative models, where the dimensionality of the latent is much smaller than the signal. This can be especially troubling when one considers the manifold hypothesis, which states that the manifold in which the signal resides, is a low-dimensional space. To mitigate these drawbacks, diffusion models in the latent space were proposed [25, 11].

The construction of the diffusion trajectory is identical to the diffusion in the pixel space, as introduced in Sec. 2. There are mainly two choices for constructing the latent diffusion: LSGM [25] proposes an end-to-end training of both the VAE and the diffusion part by posing the whole model as a hierarchical VAE, which is possible as the diffusion model themselves can be seen as a hierarchical variational VAE. Later, a simpler approach, known as LDM [11] was proposed, delineating the training into two stages. In the first stage of training LDMs, only the VAE is trained to compress the signal into a compact representation $\boldsymbol{z} \in \mathbb{R}^k$ with $k < d$

$$\boldsymbol{x} = \mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{z}), \quad \text{where} \quad \boldsymbol{z} = \mathcal{E}_{\boldsymbol{\phi}}(\boldsymbol{x}) := \mathcal{E}^{\boldsymbol{\mu}}_{\boldsymbol{\phi}}(\boldsymbol{x}) + \mathcal{E}^{\boldsymbol{\sigma}}_{\boldsymbol{\phi}}(\boldsymbol{x}) \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I}),$$

where $\mathcal{E}_\phi$ is the encoder, and $\mathcal{D}_\varphi$ is the decoder. In the second stage, using a pre-trained encoder of the VAE, the diffusion model is trained. In LDMs, a conditioning scheme was also introduced, where the network takes in another input $\boldsymbol{c}$ through cross attention [26], leading to the following training scheme

$$\min_\theta \mathbb{E}_{\boldsymbol{z}_t \sim p(\boldsymbol{z}_t|\boldsymbol{z}_0), \boldsymbol{z}_0 \sim p(\mathcal{E}(\boldsymbol{x}_0)), (\boldsymbol{x}_0, \boldsymbol{c}) \sim p(\boldsymbol{x}_0, \boldsymbol{c}) \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})} \left[\| \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t; \boldsymbol{c}) - \boldsymbol{\epsilon} \|_2^2 \right]. \tag{3.1}$$

Setting $\boldsymbol{c}$ to be the text embedding of the pre-trained CLIP [27] encoder, a text-to-image (T2I) diffusion model was constructed. Later, scaling the compute and data led to the popular stable diffusion (SD) [11, 28, 29].

Thanks to its ability to model high-resolution data efficiently, latent diffusion has become the standard of modern foundation models [29, 30, 31]. There exists another approach called cascaded diffusion [32, 10], which also enables high-resolution signal synthesis. Cascaded diffusion models work by training a low-resolution pixel diffusion as the first stage, then training several conditional super-resolution diffusion models for scaling the resolution up. While such a choice was equally popular in the early days, ranging from image [32, 10] to video [14, 33], the popularity has ceased, as it requires multiple model training as well as using multiple models at inference time.

3.2. **Diffusion Bridges and Other Formulations.**

3.2.1. *General corruptions.* Up until now, we mostly discussed standard Gaussian noising diffusion, where the terminal reference distribution becomes an isotropic normal distribution. In this section, we revisit the design choices made in standard diffusion models and explore what other choices can be made to construct different generative processes. Several other choices can be made by selecting different perturbation kernels. One popular other than Gaussian noise was to choose the forward process to be a linear corruption + noise [34, 35], where the corruption can be blurring, masking, etc. that gradually leads to the terminal distribution. Going further, the

noise component from the corruption process can also be taken away, leading to a deterministic corruption [36, 37].

3.2.2. *Direct bridges.* Diffusion models are constructed by building a specific path between the data distribution and the Gaussian distribution. This formulation suffices for generative modeling, as all we need is a reference distribution that can be easily sampled from. However, in many scientific applications, we consider *transfer* tasks, where we have two different distributions, either we have a coupling or not, and we wish to build a bridge between these two distributions π_0 and π_1.

Let us start with the easier case when we have matching pairs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{x}_0 \sim \pi_0$, $\boldsymbol{x}_1 \sim \pi_1$. To build a bridge in such case, various formulations that lead to very similar algorithms have been proposed: flow matching (FM) [38, 39], rectified flow (RF) [40], stochastic interpolants [41], etc. In these frameworks, a vector field is parametrized by a neural network $v^\theta : \mathbb{R}^d \mapsto \mathbb{R}^d$, which gradually transforms π_0 to π_1 as $t = 0 \to 1$ through an ODE

$$d\boldsymbol{x}_t = v_t^\theta(\boldsymbol{x}_t, t)\, dt$$

Interestingly, the training of v^θ can be done through a simple regression [40, 38]

$$\min_\theta \mathbb{E}_{t\sim\mathrm{Unif}(0,1),(\boldsymbol{x}_0,\boldsymbol{x}_1)\sim(\pi_0,\pi_1)} \left[\|(\boldsymbol{x}_1 - \boldsymbol{x}_0) - v_t^\theta(\boldsymbol{x}_t)\|^2\right] \quad \text{where} \quad \boldsymbol{x}_t = t\boldsymbol{x}_1 + (1-t)\boldsymbol{x}_0$$

Once the network is trained, one can simply solve the ODE defined by the vector field v^θ. In RF [40], it was further shown that one can achieve *straighter* paths by applying the reflow procedure, which corresponds to iteratively training a new RF model starting from the coupling achieved by the previous masdfodel. Interestingly, this was shown to scale even for unpaired tasks where we do not have explicit coupling. In the context of image restoration with paired data, several works proposed similar methods that enable gradual restoration from the degraded image [42, 43, 44], which were later shown to be equivalent [45] under the name direct diffusion bridges, where one can further improve restoration quality by imposing data consistency steps.

3.2.3. *Schrödinger bridge (SB).* The hardest case, but an important one, considers a mapping between π_0 and π_1 when we have unpaired samples from each distribution. To tackle this hard problem, Schrödinger bridge (SB) [46, 47, 48, 49] was introduced, which is a dynamic version of the entropy-regularized optimal transport (OT) [50] problem.

In essence, modern SB methods are linked to diffusion models by learning two nonlinear SDEs, where to run the respective SDEs, one needs to train two different diffusion models that are responsible for propagation. While these methods are versatile and can be applied to hard tasks that standard diffusion models cannot handle, such as image translation [51, 52] and atmospheric downscaling [53], these methods are hard to train robustly, and typically takes much longer to converge.

4. SAMPLING

4.1. **Solving Differential Equations.** As studied in Sec. 2, sampling from diffusion models involves solving the SDE or the ODE, which naturally requires multiple NFE with fine discretization to solve the DEs without introducing large discretization errors. For instance, DDPM [7] requires 1000 NFE, and score-SDE [5] requires 4000 NFE to achieve the best performance. Accelerating SDE solving through introducing more advanced solvers [54, 5] than a simple Euler-Maruyama discretization was explored, but with a small gain.

Later, significant attention has shifted towards ODE solving, leading to accelerated sampling that requires only 10$\sim$50 sampling steps to achieve similar sample quality [55, 56, 57, 58, 59]. It was shown that higher-order solvers, such as Heun's 2nd order method [58] and exponential integrators [56, 59] were shown to balance well the trade-off between computation and accuracy. Orthogonal to these advances, some methods aim to directly predict higher-order score functions to also estimate the curvature of the trajectory [60, 61]. The higher-order information can also be emulated by aggregating information from multiple steps during sampling, known as the multi-step method [57].

4.2. **Guided Sampling.** As the momentum towards large vision foundation models grow, where a diffusion model is trained on an extremely large corpus of data (e.g. LAION-5B [62]) it is often inconvenient and ineffective to simply train an unconditional model without conditioning. For instance, it is known that the sample quality greatly improves by limiting the possible modes by class-conditioning [63, 9]. In the modern era, the standard is to use *text* conditioning, which acts as a natural interface of the users, while being much more versatile and expressive than using class conditioning. In conditional diffusion models, the goal is to model the conditional distribution $p(\boldsymbol{x}|\boldsymbol{c})$, which involves predicting the conditional score function $\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}|\boldsymbol{c})$. Although vanilla guidance is straightforward by additionally taking in as input $\boldsymbol{c}$ for the diffusion model, a conditional diffusion model trained in this manner often learns to disregard or minimize the provided conditioning information.

To achieve effective and adjustable conditioning strength, Classifier guidance [9] utilizes a trained classifier to guide the reverse process towards a targeted mode of distribution.

$$\begin{aligned} \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}) &= \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{c}|\boldsymbol{x}_t) \\ &\approx \boldsymbol{s}_\theta(\boldsymbol{x}_t, t) + \omega \nabla_{\boldsymbol{x}_t} f_\phi(\boldsymbol{c}|\boldsymbol{x}_t), \end{aligned} \tag{4.1}$$

where f_ϕ is the classifier trained to estimate the log probability of the class label that the *noisy* input $\boldsymbol{x}_t$ belongs, and ω controls the emphasis on the conditioning, trading off diversity against quality. Notably, in this case, we cannot simply take an off-the-shelf classifier, as it is only trained on clean images without noise. Consequently, one notable limitation of classifier guidance is its dependence on a separately trained classifier. Moreover, it is known that using the gradients of the classifier may lead to adversarial gradients that push the results toward undesirable outcomes [64].

Classifier-Free Guidance (CFG)[65] addresses this issue by using only the diffusion model inference. Rewriting Bayes rule in Eq. (4.1), we have

$$\begin{aligned}\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{c}|\boldsymbol{x}_t) &= \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}) - \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) \\ &\approx \boldsymbol{s}_\theta(\boldsymbol{x}_t, t; \boldsymbol{c}) - \boldsymbol{s}_\theta(\boldsymbol{x}_t, t).\end{aligned}$$

Clearly, we can see that we can compute the gradient of the log-likelihood as in classifier guidance *without* any classifier if we have access to the conditional score function $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t; \boldsymbol{c})$. When training a diffusion model, this can be easily implemented as in Eq. (3.1) by dropping $\boldsymbol{c}$ for certain probabilities to also be able to use the unconditional score $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$. Similar to classifier guidance, we can choose a hyperparameter $\omega > 0$ to control the guidance scale:

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}) \approx \boldsymbol{s}_\theta(\boldsymbol{x}_t, t; \boldsymbol{c}) + \omega(\boldsymbol{s}_\theta(\boldsymbol{x}_t, t; \boldsymbol{c}) - \boldsymbol{s}_\theta(\boldsymbol{x}_t, t)).$$

When using guided sampling with ω-sharpening, we end up sampling from $p(\boldsymbol{x}_t)p(\boldsymbol{c}|\boldsymbol{x}_t)^\omega$, which is only the true posterior when $\omega = 1$. When $\omega = 0$, CFG reduces to unconditional sampling. In practice, in order to achieve high-quality samples, one typically sets high CFG guidance scale e.g. $\omega = 7.5$. While CFG excels at generating remarkable images aligned with the input text condition, even minor text alterations can yield entirely different outputs. Semantic Guidance [66] focused on disentangling semantic directions inherent to the model, enabling subtle control throughout the generation. On the other hand, Hong*et al.*[67] use by-products of the generation process like attention maps for guidance instead of external conditions.

4.3. **Diffusion Distillation.**

4.3.1. *Sampling acceleration.* The methods belonging to this category let us *integrate* longer paths of the PF-ODE trajectory whereas diffusion models are only capable of producing the *tangent* direction of the curved trajectory.

Diffusion models are inherently slow to sample from. The first work that belongs to the first category was proposed by Luhman and Luhman [68]

$$\min_\phi \|G_\phi(\boldsymbol{x}_T) - \boldsymbol{x}_0^\theta\|, \quad \boldsymbol{x}_0^\theta = \int_T^0 \mathrm{PFODE}(\boldsymbol{x}_T; \theta, \boldsymbol{x}_s)\, ds, \quad \boldsymbol{x}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}),$$

where G_ϕ is a new generative model that maps the noise $\boldsymbol{x}_T$ to $\boldsymbol{x}_0$ in one-step, and $\boldsymbol{x}_0^\theta$ is the *target* sample generated from the pre-trained diffusion model by deterministic sampling through the PF-ODE trajectory initialized with $\boldsymbol{x}_T$. While this method was shown to be feasible, sampling $\boldsymbol{x}_0^\theta$ to train a new ϕ is very compute-heavy. Later, it was shown that we can dissect this process into iterative distillation, where the student network learns the 2-step DDIM sampling process of the teacher network, known as progressive distillation [69]. While faster than [68], this has the downside of requiring training a new model whenever aiming for reducing the NFE by a factor of 2, each time introducing new errors.

Consistency Model (CM) is a popular distillation choice that aims to tackle these drawbacks, by designing a network $G_\phi : \mathbb{R}^d \times \mathbb{R} \mapsto \mathbb{R}^d$, which can take any $\boldsymbol{x}_t$, $t \in [0, 1]$ and map it

directory to $t = 0$ in a single step. The training proceeds by

$$\min_{\phi} \mathbb{E}\left[\|G_\phi(\boldsymbol{x}_{t-\Delta t}, t-\Delta t) - G_\phi(\boldsymbol{x}_t, t)\|\right], \quad \boldsymbol{x}_{t-\Delta t} = \int_t^{t-\Delta t} \mathrm{PFODE}(\boldsymbol{x}_t; \theta, \boldsymbol{x}_s)\, ds.$$

Optimized over all horizon, G_ϕ learns to map any x_t along the trajectory to $\boldsymbol{x}_0$ regardless of t. One downside of CM is that it lacks a way to trade off quality for speed. Consistency Trajectory Model (CTM) [70] proposes a generalization of CM and diffusion models, by designing the student network G_ϕ to take in two time conditions: the starting point of the integral, and the ending point of the integral. When these two points are the same, the network boils down to the original diffusion model. When the ending point is set to 0, CTM acts like a CM. Similar ideas with different design choices on how to distill the PF-ODE trajectory were independently proposed in TRACT [71], BOOT [72], etc. Perhaps not surprisingly, the same ideas were soon applied to LDMs [73, 74, 75].

4.3.2. *Score distillation sampling.* The methods that belong to this category started from a method called score distillation sampling (SDS) [12], where the goal is *not* to distill the reverse diffusion trajectory but to use the diffusion model as a *testing* function, analogous to the role of discriminators in GAN [1]. This is natural, as the use of denoisers as a testing function, or similarly a regularizer that pushes it towards the image manifold has been studied extensively in the past literature (e.g. RED [76], Plug-and-play prior [77]).

Consider a differentiable, parameterized function that can generate an image as an output $\boldsymbol{x} = g(\phi)$. SDS provides a way to train the parameters of ϕ to generate *plausible* $\boldsymbol{x}$s under the diffusion prior. The straightforward loss reads

$$\nabla_\phi \mathcal{L}_{\text{Diff}}\left(\theta, \boldsymbol{x} = g(\phi)\right) = \mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\underbrace{\left(\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t; \boldsymbol{c}, t\right) - \boldsymbol{\epsilon}\right)}_{\text{Noise Residual}} \underbrace{\frac{\partial \boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t; \boldsymbol{c}, t\right)}{\boldsymbol{x}_t}}_{\text{U-Net Jacobian}} \underbrace{\frac{\partial \boldsymbol{x}}{\partial \phi}}_{\text{Generator Jacobian}}\right]. \tag{4.2}$$

We see that in order to compute the gradient, we have to compute the U-Net Jacobian of the heavy diffusion model, which is cumbersome, and highly unstable especially in the low noise regime [78]. SDS proposes a surprisingly simple fix to Eq. (4.2) by setting the U-Net Jacobian as the identity matrix (skipping over the gradient)

$$\nabla_\phi \mathcal{L}_{\text{SDS}} = \mathbb{E}\left[\left(\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t; \boldsymbol{c}) - \boldsymbol{\epsilon}\right)\frac{\partial \boldsymbol{x}}{\partial \phi}\right], \tag{4.3}$$

which is much cheaper and faster to compute. Interestingly, while the choice seems arbitrary, the resulting scheme Eq. (4.3) turns out to be probability density distillation [79]. The original work of SDS employs 2D pre-trained text-conditioned diffusion models for when $g(\phi)$ is a 3D Neural Radius Field (NeRF) [80] renderer. Building upon SDS, several subsequent works have been developed, focusing not only on the per-scene optimization of 3D objects [13, 81, 82, 83], but also on text-to-3D video synthesis [84] and image editing [85, 86, 87]. Despite its widespread application, SDS is recognized for prominent concerns such as over-saturation, over-smoothing and a lack of diversity.

Variational Score Distillation (VSD) [13] tackles this issue by introducing the VSD loss. Contrary to SDS, which models the 3D parameter ϕ as a constant, VSD treats it as a random variable. In order to model the 3D distribution, VSD employs particle-based variational inference and maintains a set of 3D parameters as particles $\{\phi_i\}_i$. For the implementation, another diffusion model with low-rank adaptation (LoRA), trained during the NeRF optimization process, is utilized. On the other hand, Collaborative Score Distillation [86] treats multiple samples in the Stein Variational Gradient Descent [88] update. It adjusts the score function to enhance the consistency among a set of images simultaneously. Additionally, many studies are emerging to analyze and improve the limitations of SDS term. Noise Free Score Distillation [89] decomposes the score into interpretable components and redesigns the term to exclude noise, thereby preventing undesirable noise distillation during optimization. Yu *et al.*propose Classifier Score Distillation [90], based on the finding that classifier score $\nabla_{\boldsymbol{x}_t} \log q(y|\boldsymbol{x}_t)$ alone is sufficient for effective generation.

5. APPLICATIONS

5.1. **Text-to-x Foundation Models.**

5.1.1. *Text-to-Image Generation.* Text-to-image (T2I) generation is the task of generating an image that corresponds to a provided descriptive text. Previous T2I models, working in pixel space, have improved both sample fidelity and image-text alignment through the integration of CFG and pre-trained text encoders into the T2I process [91, 10].

The emergence of LDM [11] has accelerated the progress of T2I models. A notable framework in this domain is SD [11], which utilizes pre-trained VQGAN [92] for latent representation and introduces cross-attention for diverse conditioning. DALL-E2 [93] leverages a different approach, training a diffusion model in the CLIP [27] embedding space.

While T2I models such as Imagen and DALL-E2 excel in achieving remarkable image fidelity and caption alignment, they often lack the capability to provide fine-grained control over spatial structure. In response, several works have utilized more specific conditions to attain higher fidelity and precise control. Make-A-Scene [94] and SpaText [95] leverage segmentation masks to guide the generation process, while GLIGEN [96] enables a pretrained T2I diffusion model to be conditioned on bounding boxes. Furthermore, Zhang *et al.* [97] present ControlNet, a neural network architecture that links a trainable copy and the original frozen model through a specialized convolution layer. This layer initializes weights to zeros and does not add noise during the fine-tuning process.

5.1.2. *Text-to-3D Generation.* Given the success of numerous diffusion models in generating high-quality and realistic 2D images, there is a growing interest in extending these models to the 3D field. One approach in this direction is training a diffusion model using 3D data [98, 99, 100]. However, these methods are constrained by the requirement for modality-specific data.

DreamFusion [12] addresses this limitation by optimizing a 3D representation to ensure that the rendered image, from any viewpoint, maintains a high likelihood as assessed by the diffusion model, given the text prompt. Latent-NeRF [83] extends the score distillation framework to a

condensed latent space, thereby boosting computational efficiency. ED-NeRF [101] enhances the latent generation process by introducing a refinement layer and extending the Delta Denoising Score [85] into the 3D domain. While these approaches enable the utilization of the vast data domain of 2D diffusion models, the generated 3D assets often suffer from inconsistencies in geometry formation due to the lack of awareness of the camera view. To address this, Zero-1-to-3 [102] finetunes a pretrained T2I diffusion model to incorporate camera pose conditioning, enabling zero-shot novel view synthesis and 3D reconstruction from a single image. Additionally, there are models that utilize CLIP [27] to align each view of the 3D representation model with the corresponding text [103, 104, 105].

Recently, with the emergence of 3D Gaussian Splatting as a novel representation method for 3D scenes, there has been active research into utilizing diffusion models as priors for training Gaussian Splatting representation [106, 107].

5.1.3. *Text-to-Video Generation.* Video Diffusion Models (VDMs) originated with the work of Ho *et al.* [15], integrated diffusion models into video generation tasks using 3D U-Net architecture. Subsequently, Make-A-Video [33] proposes spatio-temporally factorized diffusion model, which is built upon a pretrained T2I model. This improvement is achieved through the utilization of pseudo-3D convolution and temporal attention layers. On the other hand, Imagen Video [14] is a text-conditional video generation system that relies on cascaded video diffusion models. The utilization of configurations informed by recent discoveries, such as employing a frozen T5 text encoder and classifier-free guidance, enhances its performance. Blattman *et al.* [16] and Zhou *et al.* [108] apply the LDM paradigm to high-resolution video generation and improve training efficiency. Building upon latent-based VDMs, Tune-A-Video [109] proposes a one-shot video tuning method that does not necessitate large-scale video datasets. Show-1 [110] combines the strengths and addresses the weaknesses of both pixel-based and latent-based VDMs, resulting in notable performance improvements.

5.2. **Inverse Problems.** Given

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}, \quad \boldsymbol{y} \in \mathbb{R}^n, \boldsymbol{x} \in \mathbb{R}^d, \boldsymbol{A} \in \mathbb{R}^{d\times n}, \boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \sigma_y^2 \boldsymbol{I}), \tag{5.1}$$

the goal of inverse problems is to infer the unknown signal $\boldsymbol{x}$ from the degraded measurement $\boldsymbol{y}$. One canonical way to solve the problem is through posterior sampling from $p(\boldsymbol{x}|\boldsymbol{y}) \propto p(\boldsymbol{x})p(\boldsymbol{y}|\boldsymbol{x})$, where the likelihood $p(\boldsymbol{y}|\boldsymbol{x})$ is given from Eq. (5.1), and one needs to specify the prior $p(\boldsymbol{x})$. Diffusion model-based inverse problem solvers (DIS) use the diffusion prior by using a plug-in approximation $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) \approx \boldsymbol{s}_{\theta^*}(\boldsymbol{x}_t, t)$, and devise different ways of incorporating the likelihood during the sampling process. Earlier methods used alternating projections in between the unconditional denoising steps of the diffusion model [5, 111, 112, 113, 114]. Later, attempts to approximate the intractable time-dependent log-likelihood $p(\boldsymbol{y}|\boldsymbol{x}_t)$ were proposed. Score-ALD [115] uses

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \approx A^\top \frac{(A\boldsymbol{x}_t - \boldsymbol{y})}{\sigma_y^2 + \gamma_t^2},$$

where γ_t is an annealing hyper-parameter. The family of DDRM [116, 117] uses

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \approx \frac{\boldsymbol{y} - \boldsymbol{x}_t}{|\sigma_y^2 - \sigma_t^2|},$$

where σ_t is the noise level of the diffusion at time t, for the case of $\boldsymbol{A} = \boldsymbol{I}$. For a general $\boldsymbol{A}$, one computes singular value decomposition (SVD) to weight the spectral components differently according to the noise level in that specific component. Diffusion Posterior Sampling (DPS) uses

$$p(\boldsymbol{y}|\boldsymbol{x}_t) = \int p(\boldsymbol{y}|\boldsymbol{x}_0)p(\boldsymbol{x}_0|\boldsymbol{x}_t)d\boldsymbol{x}_0 \approx p(\boldsymbol{y}|\boldsymbol{x}_0 = \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]),$$

by leveraging the Jensen's approximation. DPS is a general framework that can be used to solve a fully general class of inverse problems including non-linear operators and various noise types. Variants include ΠGDM [118] that uses a Gaussian approximation for $p(\boldsymbol{x}_0|\boldsymbol{x}_t)$, and MCG [119] that additionally imposes projections onto the measurement subspace.

DIS was extended to LDMs, which are typically more useful for high-resolution image reconstruction leveraging text guidance [120, 121, 122, 123, 124]. DIS was also extended to more challenging cases, including when the operator $\boldsymbol{A}$ is blind [125, 126, 127]; when the signal to reconstruct is 3D, but the prior is only modeled in 2D [128, 129]; and when there is a mismatch between the training distribution and the testing distribution [130].

5.3. **Image editing with Text-to-image Diffusion Models.** The objective of text-driven image editing is to align with the content specified in a target prompt while also integrating the structure and overall styles of an input image. DiffusionCLIP [131] enables image manipulation by finetuning the score function during the reverse diffusion process, guided by a CLIP loss that manages the attributes outlined in the text prompt. Yang *et al.* [132] leverages both CLIP loss and contrastive loss to enable zero-shot style transfer. The widespread use of LDMs has spurred extensive exploration into various research. Prompt-to-Prompt [133] and Pix2Pix-zero [134] propose preserving certain content from the source image by leveraging the cross-attention maps. On the other hand, Plug and Play Diffusion [135] injects spatial features from the decoder and their self-attention map. This enables precise controlled image translation over the generated shape and layout. Combining these methods with the inversion of real images [136, 137] shows enhanced editing performance.

6. Conclusion

In this work, we reviewed the theory of diffusion models, their variations in the trajectory and the sampling process, and their widespread applications. While there have been tremendous advances in the field, benefitting not only from the technical advances but also from the sheer scaling of data and compute, unsolved questions and applications remain. Is denoising trajectory the optimal generative path? What is the better architecture, U-Nets or Transformers? Will diffusion models start to prevail in the language domain, as they do in vision? The answers to the posed questions will not only push the boundaries of what is technically possible but also deepen our understanding of the underlying principles that make diffusion models so effective.

REFERENCES

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative adversarial nets*. Advances in neural information processing systems, **27**, 2014.

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

[3] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. In Advances in Neural Information Processing Systems.

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

[5] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. *Score-Based Generative Modeling through Stochastic Differential Equations*. In 9th International Conference on Learning Representations, ICLR, 2021.

[6] 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.

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

[8] Aapo Hyvärinen and Peter Dayan. *Estimation of non-normalized statistical models by score matching*. Journal of Machine Learning Research, **6**(4), 2005.

[9] Prafulla Dhariwal and Alexander Quinn Nichol. *Diffusion Models Beat GANs on Image Synthesis*. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, Advances in Neural Information Processing Systems, 2021.

[10] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. *Photorealistic text-to-image diffusion models with deep language understanding*. Advances in neural information processing systems, **35**:36479–36494, 2022.

[11] 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.

[12] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. *DreamFusion: Text-to-3D using 2D Diffusion*. arXiv, 2022.

[13] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. *Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation*. Advances in Neural Information Processing Systems, **36**, 2024.

[14] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. *Imagen video: High definition video generation with diffusion models*. arXiv preprint arXiv:2210.02303, 2022.

[15] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. *Video diffusion models*. Advances in Neural Information Processing Systems, **35**:8633–8646, 2022.

[16] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. *Align your latents: High-resolution video synthesis with latent diffusion models*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 22563–22575, 2023.

[17] Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. *Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models*. arXiv preprint arXiv:2312.13763, 2023.

[18] Chin-Wei Huang, Jae Hyun Lim, and Aaron Courville. *A Variational Perspective on Diffusion-Based Generative Models and Score Matching*. arXiv preprint arXiv:2106.02808, 2021.

[19] Brian DO Anderson. *Reverse-time diffusion equation models*. Stochastic Processes and their Applications, **12**(3):313–326, 1982.

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

[21] Bradley Efron. *Tweedie's formula and selection bias*. Journal of the American Statistical Association, **106**(496):1602–1614, 2011.

[22] Kwanyoung Kim and Jong Chul Ye. *Noise2Score: Tweedie's Approach to Self-Supervised Image Denoising without Clean Images*. Advances in Neural Information Processing Systems, **34**, 2021.

[23] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. *simple diffusion: End-to-end diffusion for high resolution images*. In International Conference on Machine Learning, pages 13213–13232. PMLR, 2023.

[24] Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. *Scalable High-Resolution Pixel-Space Image Synthesis with Hourglass Diffusion Transformers*. arXiv preprint arXiv:2401.11605, 2024.

[25] Arash Vahdat, Karsten Kreis, and Jan Kautz. *Score-based generative modeling in latent space*. Advances in neural information processing systems, **34**:11287–11302, 2021.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. *Attention is all you need*. Advances in neural information processing systems, **30**, 2017.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. *Learning transferable visual models from natural language supervision*. In International conference on machine learning, pages 8748–8763. PMLR, 2021.

[28] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. *Sdxl: Improving latent diffusion models for high-resolution image synthesis*. arXiv preprint arXiv:2307.01952, 2023.

[29] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. *Scaling Rectified Flow Transformers for High-Resolution Image Synthesis*. arXiv preprint arXiv:2403.03206, 2024.

[30] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. *PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis*. arXiv preprint arXiv:2310.00426, 2023.

[31] Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. *Playground v2. 5: Three Insights towards Enhancing Aesthetic Quality in Text-to-Image Generation*. arXiv preprint arXiv:2402.17245, 2024.

[32] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. *Cascaded diffusion models for high fidelity image generation*. Journal of Machine Learning Research, **23**(47):1–33, 2022.

[33] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. *Make-a-video: Text-to-video generation without text-video data*. arXiv preprint arXiv:2209.14792, 2022.

[34] Giannis Daras, Mauricio Delbracio, Hossein Talebi, Alexandros G Dimakis, and Peyman Milanfar. *Soft diffusion: Score matching for general corruptions*. arXiv preprint arXiv:2209.05442, 2022.

[35] Sangyun Lee, Hyungjin Chung, Jaehyeon Kim, and Jong Chul Ye. *Progressive deblurring of diffusion models for coarse-to-fine image synthesis*. arXiv preprint arXiv:2207.11192, 2022.

[36] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. *Cold diffusion: Inverting arbitrary image transforms without noise*. Advances in Neural Information Processing Systems, **36**, 2024.

[37] Severi Rissanen, Markus Heinonen, and Arno Solin. *Generative modelling with inverse heat dissipation*. arXiv preprint arXiv:2206.13397, 2022.

[38] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. *Flow matching for generative modeling*. arXiv preprint arXiv:2210.02747, 2022.

[39] Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky Chen. *Multisample flow matching: Straightening flows with minibatch couplings*. arXiv preprint arXiv:2304.14772, 2023.

[40] Xingchao Liu, Chengyue Gong, and Qiang Liu. *Flow straight and fast: Learning to generate and transfer data with rectified flow*. arXiv preprint arXiv:2209.03003, 2022.

[41] Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. *Stochastic interpolants: A unifying framework for flows and diffusions*. arXiv preprint arXiv:2303.08797, 2023.

[42] Mauricio Delbracio and Peyman Milanfar. *Inversion by Direct Iteration: An Alternative to Denoising Diffusion for Image Restoration*. arXiv preprint arXiv:2303.11435, 2023.

[43] Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. *I^2SB: Image-to-Image Schrödinger Bridge*. In International conference on machine learning. PMLR, 2023.

[44] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. *Image Restoration with Mean-Reverting Stochastic Differential Equations*. In International conference on machine learning. PMLR, 2023.

[45] Hyungjin Chung, Jeongsol Kim, and Jong Chul Ye. *Direct diffusion bridge using data consistency for inverse problems*. Advances in Neural Information Processing Systems, **36**, 2024.

[46] Christian Léonard. *A survey of the schrödinger problem and some of its connections with optimal transport*. arXiv preprint arXiv:1308.0215, 2013.

[47] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. *Diffusion Schrödinger bridge with applications to score-based generative modeling*. Advances in Neural Information Processing Systems, **34**:17695–17709, 2021.

[48] Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. *Deep generative learning via schrödinger bridge*. In International conference on machine learning, pages 10794–10804. PMLR, 2021.

[49] Guan-Horng Liu, Tianrong Chen, Oswin So, and Evangelos Theodorou. *Deep generalized schrödinger bridge*. Advances in Neural Information Processing Systems, **35**:9374–9388, 2022.

[50] Cédric Villani et al. *Optimal transport: old and new*, volume **338**. Springer, 2009.

[51] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. *Dual diffusion implicit bridges for image-to-image translation*. arXiv preprint arXiv:2203.08382, 2022.

[52] Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. *Diffusion Schrödinger bridge matching*. Advances in Neural Information Processing Systems, **36**, 2024.

[53] Tobias Bischoff and Katherine Deck. *Unpaired downscaling of fluid flows with diffusion bridges*. arXiv preprint arXiv:2305.01822, 2023.

[54] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. *Gotta Go Fast When Generating Data with Score-Based Models*. arXiv preprint arXiv:2105.14080, 2021.

[55] Jiaming Song, Chenlin Meng, and Stefano Ermon. *Denoising Diffusion Implicit Models*. In 9th International Conference on Learning Representations, ICLR, 2021.

[56] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. *DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps*. In Advances in Neural Information Processing Systems, 2022.

[57] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. *Pseudo Numerical Methods for Diffusion Models on Manifolds*. In International Conference on Learning Representations, 2022.

[58] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. *Elucidating the Design Space of Diffusion-Based Generative Models*. In Proc. NeurIPS, 2022.

[59] Qinsheng Zhang and Yongxin Chen. *Fast sampling of diffusion models with exponential integrator*. arXiv preprint arXiv:2204.13902, 2022.

[60] Chenlin Meng, Yang Song, Wenzhe Li, and Stefano Ermon. *Estimating high order gradients of the data distribution by denoising*. Advances in Neural Information Processing Systems, **34**:25359–25369, 2021.

[61] Tim Dockhorn, Arash Vahdat, and Karsten Kreis. *Genie: Higher-order denoising diffusion solvers*. Advances in Neural Information Processing Systems, **35**:30150–30166, 2022.

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. *Laion-5b: An open large-scale dataset for training next generation image-text models*. Advances in Neural Information Processing Systems, **35**:25278–25294, 2022.

[63] Andrew Brock, Jeff Donahue, and Karen Simonyan. *Large Scale GAN Training for High Fidelity Natural Image Synthesis*. In International Conference on Learning Representations (ICLR), 2019.

[64] Bahjat Kawar, Roy Ganz, and Michael Elad. *Enhancing diffusion-based image synthesis with robust classifier guidance*. arXiv preprint arXiv:2208.08664, 2022.

[65] Jonathan Ho and Tim Salimans. *Classifier-Free Diffusion Guidance*. In NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications, 2021.

[66] Manuel Brack, Felix Friedrich, Dominik Hintersdorf, Lukas Struppek, Patrick Schramowski, and Kristian Kersting. *SEGA: Instructing text-to-image models using semantic guidance*. Advances in Neural Information Processing Systems, **36**, 2024.

[67] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. *Improving sample quality of diffusion models using self-attention guidance*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 7462–7471, 2023.

[68] Eric Luhman and Troy Luhman. *Knowledge distillation in iterative generative models for improved sampling speed*. arXiv preprint arXiv:2101.02388, 2021.

[69] Tim Salimans and Jonathan Ho. *Progressive Distillation for Fast Sampling of Diffusion Models*. In International Conference on Learning Representations, 2022.

[70] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. *Consistency trajectory models: Learning probability flow ode trajectory of diffusion*. arXiv preprint arXiv:2310.02279, 2023.

[71] David Berthelot, Arnaud Autef, Jierui Lin, Dian Ang Yap, Shuangfei Zhai, Siyuan Hu, Daniel Zheng, Walter Talbott, and Eric Gu. *Tract: Denoising diffusion models with transitive closure time-distillation*. arXiv preprint arXiv:2303.04248, 2023.

[72] Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Lingjie Liu, and Joshua M Susskind. *Boot: Data-free distillation of denoising diffusion models with bootstrapping*. In ICML 2023 Workshop on Structured Probabilistic Inference {\&} Generative Modeling, 2023.

[73] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. *Latent consistency models: Synthesizing high-resolution images with few-step inference*. arXiv preprint arXiv:2310.04378, 2023.

[74] Xiang Wang, Shiwei Zhang, Han Zhang, Yu Liu, Yingya Zhang, Changxin Gao, and Nong Sang. *Videolcm: Video latent consistency model*. arXiv preprint arXiv:2312.09109, 2023.

[75] Junsong Chen, Yue Wu, Simian Luo, Enze Xie, Sayak Paul, Ping Luo, Hang Zhao, and Zhenguo Li. *PIXART-{\delta}: Fast and Controllable Image Generation with Latent Consistency Models*. arXiv preprint arXiv:2401.05252, 2024.

[76] Yaniv Romano, Michael Elad, and Peyman Milanfar. *The little engine that could: Regularization by denoising (RED)*. SIAM Journal on Imaging Sciences, **10**(4):1804–1844, 2017.

[77] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. *Plug-and-play priors for model based reconstruction*. In 2013 IEEE global conference on signal and information processing, pages 945–948. IEEE, 2013.

[78] Dongjun Kim, Seungjae Shin, Kyungwoo Song, Wanmo Kang, and Il-Chul Moon. *Soft truncation: A universal training technique of score-based diffusion model for high precision score estimation*. International conference on machine learning, 2022.

[79] Aaron Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George Driessche, Edward Lockhart, Luis Cobo, Florian Stimberg, et al. *Parallel wavenet: Fast high-fidelity speech synthesis*. In International conference on machine learning, pages 3918–3926. PMLR, 2018.

[80] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. *Nerf: Representing scenes as neural radiance fields for view synthesis*. Communications of the ACM, **65**(1):99–106, 2021.

[81] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. *Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 22246–22256, 2023.

[82] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. *Magic3d: High-resolution text-to-3d content creation*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 300–309, 2023.

[83] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. *Latent-nerf for shape-guided generation of 3d shapes and textures*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12663–12673, 2023.

[84] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. *Text-to-4d dynamic scene generation*. arXiv preprint arXiv:2301.11280, 2023.

[85] Amir Hertz, Kfir Aberman, and Daniel Cohen-Or. *Delta denoising score*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2328–2337, 2023.

[86] Subin Kim, Kyungmin Lee, June Suk Choi, Jongheon Jeong, Kihyuk Sohn, and Jinwoo Shin. *Collaborative Score Distillation for Consistent Visual Editing*. In Thirty-seventh Conference on Neural Information Processing Systems, 2023.

[87] Hyelin Nam, Gihyun Kwon, Geon Yeong Park, and Jong Chul Ye. *Contrastive Denoising Score for Text-guided Latent Diffusion Image Editing*. arXiv preprint arXiv:2311.18608, 2023.

[88] Qiang Liu and Dilin Wang. *Stein variational gradient descent: A general purpose bayesian inference algorithm*. Advances in neural information processing systems, **29**, 2016.

[89] Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. *Noise-free Score Distillation*. In The Twelfth International Conference on Learning Representations, 2024.

[90] Xin Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, and XIAOJUAN QI. *Text-to-3D with Classifier Score Distillation*. In The Twelfth International Conference on Learning Representations, 2024.

[91] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. *Glide: Towards photorealistic image generation and editing with text-guided diffusion models*. arXiv preprint arXiv:2112.10741, 2021.

[92] Patrick Esser, Robin Rombach, and Bjorn Ommer. *Taming transformers for high-resolution image synthesis*. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12873–12883, 2021.

[93] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. *Hierarchical text-conditional image generation with clip latents*. arXiv preprint arXiv:2204.06125, 2022.

[94] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. *Make-a-scene: Scene-based text-to-image generation with human priors*. In European Conference on Computer Vision, pages 89–106. Springer, 2022.

[95] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. *Spatext: Spatio-textual representation for controllable image generation*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 18370–18380, 2023.

[96] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. *Gligen: Open-set grounded text-to-image generation*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 22511–22521, 2023.

[97] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. *Adding conditional control to text-to-image diffusion models*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3836–3847, 2023.

[98] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. *3dgen: Triplane latent diffusion for textured mesh generation*. arXiv preprint arXiv:2303.05371, 2023.

[99] Heewoo Jun and Alex Nichol. *Shap-e: Generating conditional 3d implicit functions*. arXiv preprint arXiv:2305.02463, 2023.

[100] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. *Point-e: A system for generating 3d point clouds from complex prompts*. arXiv preprint arXiv:2212.08751, 2022.

[101] JangHo Park, Gihyun Kwon, and Jong Chul Ye. *ED-NeRF: Efficient Text-Guided Editing of 3D Scene With Latent Space NeRF*. In The Twelfth International Conference on Learning Representations, 2024.

[102] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. *Zero-1-to-3: Zero-shot one image to 3d object*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9298–9309, 2023.

[103] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. *Zero-shot text-guided object generation with dream fields*. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 867–876, 2022.

[104] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. *Text2mesh: Text-driven neural stylization for meshes*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 13492–13502, 2022.

[105] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. *Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20908–20918, 2023.

[106] Taoran Yi, Jiemin Fang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. *Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors*. arXiv preprint arXiv:2310.08529, 2023.

[107] Zilong Chen, Feng Wang, and Huaping Liu. *Text-to-3d using gaussian splatting*. arXiv preprint arXiv:2309.16585, 2023.

[108] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. *Magicvideo: Efficient video generation with latent diffusion models*. arXiv preprint arXiv:2211.11018, 2022.

[109] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. *Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 7623–7633, 2023.

[110] David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. *Show-1: Marrying pixel and latent diffusion models for text-to-video generation*. arXiv preprint arXiv:2309.15818, 2023.

[111] Zahra Kadkhodaie and Eero P Simoncelli. *Stochastic Solutions for Linear Inverse Problems using the Prior Implicit in a Denoiser*. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, Advances in Neural Information Processing Systems, 2021.

[112] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. *Come-Closer-Diffuse-Faster: Accelerating Conditional Diffusion Models for Inverse Problems through Stochastic Contraction*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[113] Hyungjin Chung and Jong Chul Ye. *Score-based diffusion models for accelerated MRI*. Medical Image Analysis, page 102479, 2022.

[114] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. *Solving Inverse Problems in Medical Imaging with Score-Based Generative Models*. In International Conference on Learning Representations, 2022.

[115] Ajil Jalal, Marius Arvinte, Giannis Daras, Eric Price, Alexandros G Dimakis, and Jonathan Tamir. *Robust compressed sensing mri with deep generative priors*. Advances in Neural Information Processing Systems, **34**, 2021.

[116] Bahjat Kawar, Gregory Vaksman, and Michael Elad. *Stochastic Image Denoising by Sampling From the Posterior Distribution*. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pages 1866–1875, October 2021.

[117] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. *Denoising Diffusion Restoration Models*. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, Advances in Neural Information Processing Systems, 2022.

[118] Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. *Pseudoinverse-Guided Diffusion Models for Inverse Problems*. In International Conference on Learning Representations, 2023.

[119] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. *Improving Diffusion Models for Inverse Problems using Manifold Constraints*. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, Advances in Neural Information Processing Systems, 2022.

[120] Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alexandros G Dimakis, and Sanjay Shakkottai. *Solving Linear Inverse Problems Provably via Posterior Sampling with Latent Diffusion Models*. arXiv preprint arXiv:2307.00619, 2023.

[121] Hyungjin Chung, Jong Chul Ye, Peyman Milanfar, and Mauricio Delbracio. *Prompt-tuning latent diffusion models for inverse problems*. arXiv preprint arXiv:2310.01110, 2023.

[122] Litu Rout, Yujia Chen, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. *Beyond First-Order Tweedie: Solving Inverse Problems using Latent Diffusion*. arXiv preprint arXiv:2312.00852, 2023.

[123] Jeongsol Kim, Geon Yeong Park, Hyungjin Chung, and Jong Chul Ye. *Regularization by Texts for Latent Diffusion Inverse Solvers*. arXiv preprint arXiv:2311.15658, 2023.

[124] Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. *Solving inverse problems with latent diffusion models via hard data consistency*. arXiv preprint arXiv:2307.08123, 2023.

[125] Hyungjin Chung, Jeongsol Kim, Sehui Kim, and Jong Chul Ye. *Parallel Diffusion Models of Operator and Image for Blind Inverse Problems*. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023.

[126] Naoki Murata, Koichi Saito, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Yuki Mitsufuji, and Stefano Ermon. *GibbsDDRM: A Partially Collapsed Gibbs Sampler for Solving Blind Inverse Problems with Denoising Diffusion Restoration*. In International conference on machine learning. PMLR, 2023.

[127] Charles Laroche, Andrés Almansa, and Eva Coupete. *Fast diffusion em: a diffusion model for blind inverse problems with application to deconvolution*. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 5271–5281, 2024.

[128] Hyungjin Chung, Dohoon Ryu, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. *Solving 3D Inverse Problems using Pre-trained 2D Diffusion Models*. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023.

[129] Suhyeon Lee, Hyungjin Chung, Minyoung Park, Jonghyuk Park, Wi-Sun Ryu, and Jong Chul Ye. *Improving 3D imaging with pre-trained perpendicular 2D diffusion models*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 10710–10720, 2023.

[130] Riccardo Barbano, Alexander Denker, Hyungjin Chung, Tae Hoon Roh, Simon Arrdige, Peter Maass, Bangti Jin, and Jong Chul Ye. *Steerable conditional diffusion for out-of-distribution adaptation in imaging inverse problems*. arXiv preprint arXiv:2308.14409, 2023.

[131] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. *Diffusionclip: Text-guided diffusion models for robust image manipulation*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2426–2435, 2022.

[132] Serin Yang, Hyunmin Hwang, and Jong Chul Ye. *Zero-shot contrastive loss for text-guided diffusion image style transfer*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 22873–22882, 2023.

[133] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. *Prompt-to-prompt image editing with cross attention control*. arXiv preprint arXiv:2208.01626, 2022.

[134] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. *Zero-shot image-to-image translation*. In ACM SIGGRAPH 2023 Conference Proceedings, pages 1–11, 2023.

[135] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. *Plug-and-play diffusion features for text-driven image-to-image translation*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1921–1930, 2023.

[136] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. *Null-text inversion for editing real images using guided diffusion models*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6038–6047, 2023.

[137] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. *An edit friendly ddpm noise space: Inversion and manipulations*. arXiv preprint arXiv:2304.06140, 2023.