Progressive Deblurring of Diffusion Models for Coarse-to-Fine Image Synthesis

Sangyun Lee
Soongsil University
ml.swlee@gmail.com

Hyungjin Chung
Dept. of Bio and Brain Engineering
KAIST, Korea
hj.chung@kaist.ac.kr

Jaehyeon Kim
Kakao Enterprise
jay.xyz@kakaoenterprise.com

Jong Chul Ye
Kim Jaechul Graduate School of AI
KAIST, Korea
jong.ye@kaist.ac.kr

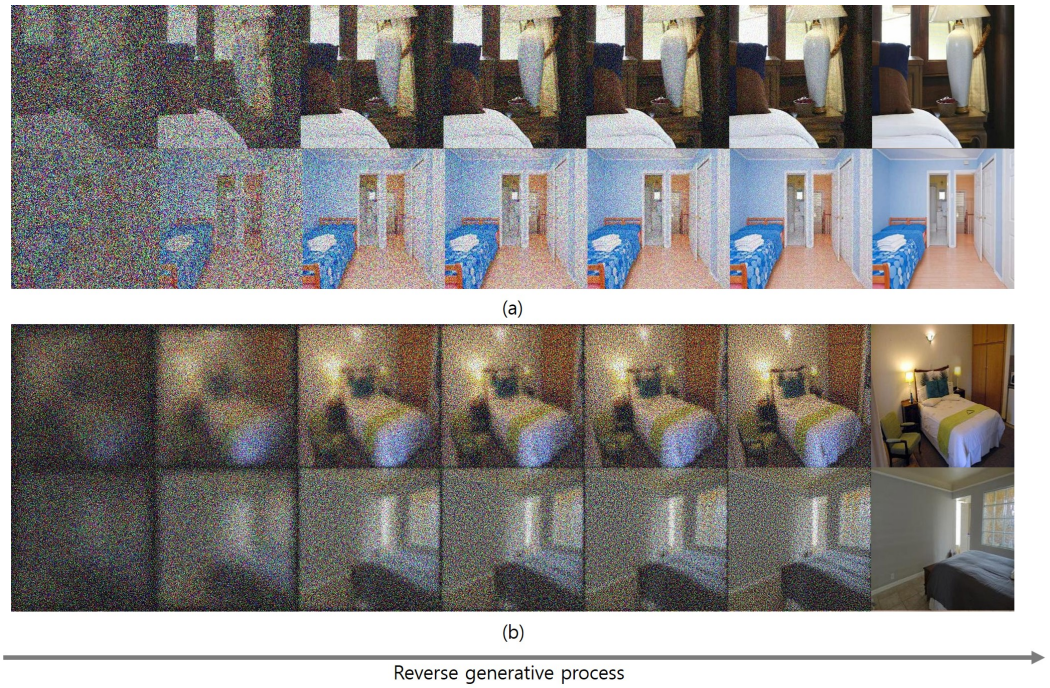

Figure 1: Reverse generative processes of two different diffusion models. (a) Previous diffusion models generate images by gradually strengthening signals. (b) The proposed method synthesizes images through progressive deblurring in a coarse-to-fine manner.

Abstract

Recently, diffusion models have shown remarkable results in image synthesis by gradually removing noise and amplifying signals. Although the simple generative process surprisingly works well, is this the best way to generate image data? For instance, despite the fact that human perception is more sensitive to the low-frequencies of an image, diffusion models themselves do not consider any relative importance of each frequency component. Therefore, to incorporate the inductive

bias for image data, we propose a novel generative process that synthesizes images in a coarse-to-fine manner. First, we generalize the standard diffusion models by enabling diffusion in a rotated coordinate system with different velocities for each component of the vector. We further propose a *blur diffusion* as a special case, where each frequency component of an image is diffused at different speeds. Specifically, the proposed blur diffusion consists of a forward process that blurs an image and adds noise gradually, after which a corresponding reverse process deblurs an image and removes noise progressively. Experiments show that proposed model outperforms the previous method in FID on LSUN bedroom and church datasets. Code is available at https://github.com/sangyun884/blur-diffusion.

1 Introduction

Diffusion models recently have shown remarkable results in image synthesis. After the initial development by Sohl-Dickstein et al. [2015], diffusion models have been rapidly improved [Ho et al., 2020, Dhariwal and Nichol, 2021, Song and Ermon, 2019, Song et al., 2020] to the point that they achieve superior results than GANs in both fidelity and diversity. Since these models offer better mode coverage, they are widely used in various tasks such as ImageNet generation [Dhariwal and Nichol, 2021], super resolution [Li et al., 2022, Saharia et al., 2021], text-conditional generation [Nichol et al., 2021, Ramesh et al., 2022], video generation [Ho et al., 2022b], etc.

Since a forward process of diffusion models attenuates signals by adding noise progressively, a reverse process generates data by gradually removing noise and amplifying signals. Although this formulation gives a great simplicity (e.g., no need to deal with a covariance matrix) and surprisingly works well, it may not be the best way to generate image data. For instance, despite the fact that human perception is more sensitive to the low-frequencies of an image, diffusion models themselves do not consider any relative importance of each frequency component.

To incorporate the inductive bias for image data, several methods have been suggested to focus on coarse patterns of an image to improve the perceptual quality of generated samples. For instance, diffusion models are usually trained on a re-weighted variational lower bound [Ho et al., 2020], which emphasizes the global consistency and coarse level pattern of images and gives less focus on imperceptible details [Kingma et al., 2021]. The performance of diffusion models can be greatly improved by adopting a coarse-to-fine strategy, where a low-resolution image is generated first and then upsampled by separate diffusion upsamplers [Ho et al., 2022a, Dhariwal and Nichol, 2021, Nichol and Dhariwal, 2021, Nichol et al., 2021, Ramesh et al., 2022]. By explicitly partitioning the generative process into the stage of generating coarse structure and the stages of adding details, these models are capable of producing convincing images, especially at high-resolution.

However, dividing into the predetermined number of stages is somewhat arbitrary and requires learning separate upsampler for each stage. In this paper, we propose a novel generative process that synthesizes images in a coarse-to-fine manner. Our model does not require any upsamplers or separate stages. Instead, we generalize the standard diffusion models by enabling diffusion in a rotated coordinate system with different velocities for each component of the vector. We further propose a *blur diffusion* as a special case of it, where each frequency component of an image is diffused at different speeds. In particular, our blur diffusion consists of a forward process that blurs an image and adds noise gradually and a corresponding reverse process that deblurs an image and removes noise progressively. Experiments show that proposed model outperforms the previous method in FID on LSUN bedroom and church datasets (64×64). We summarize our contributions as follows:

- We generalize the previous diffusion models by enabling diffusion in a rotated coordinate system with different velocities for each component of the vector.
- We propose the blur diffusion as a special case of the generalized diffusion models, where a model generates an image in a coarse-to-fine manner by progressive deblurring followed by denoising.
- Experiments show that proposed model outperforms the standard diffusion model in FID on LSUN bedroom and church datasets (64×64).

2 Background

In this section, we briefly overview the variance preserving diffusion models [Song et al., 2020]. For each training data $\mathbf{x}_0 \sim q_0(\mathbf{x})$, a forward process is defined from the following Markov chain:

$$\mathbf{x}_i = \sqrt{1-\beta_i}\mathbf{x}_{i-1} + \sqrt{\beta_i}\mathbf{z}_i, \;\; i = 1, ..., N \tag{1}$$

where $z_i \sim \mathcal{N}(0, \mathbf{I})$ and $\{\beta_i\}_{i=1}^N$ is a pre-defined noise schedule. Another way to see Eq. (1) is:

$$\mathbf{x}_i = \mathbf{x}_{i-1} - (1 - \sqrt{1-\beta_i})\mathbf{x}_{i-1} + \sqrt{\beta_i}\mathbf{z}_i, \tag{2}$$

which means that each step of the forward process consists of attenuating signals holistically and adding Gaussian noise. It is noteworthy that $q(\mathbf{x}_i|\mathbf{x}_0)$ is also Gaussian distribution and can be written in a closed form, allowing efficient training. Specifically, using the notation $\alpha_i = 1 - \beta_i$ and $\bar{\alpha}_i = \prod_{j=1}^{i} \alpha_i$, we have

$$q(\mathbf{x}_i|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_i; \sqrt{\bar{\alpha}_i}\mathbf{x}_0, (1-\bar{\alpha}_i)^2\mathbf{I}). \tag{3}$$

To generate clean images, we need to invert the noising process using sampling methods, which requires estimating the time-conditional score function $\nabla_{\mathbf{x}} \log q_i(\mathbf{x})$ where $q_i(\mathbf{x}) = \int q(\mathbf{x}_i|\mathbf{x}_0)q_0(\mathbf{x}_0)d\mathbf{x}_0$. One way to estimate the score function is to minimize the denoising score matching objective [Vincent, 2011]:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_i\{\lambda(i)\mathbb{E}_{q_0(\mathbf{x})}\mathbb{E}_{q(\mathbf{x}_i|\mathbf{x}_0)}[||\mathbf{s}_\theta(\mathbf{x}_i, i) - \nabla_{\mathbf{x}} \log q_i(\mathbf{x}_i|\mathbf{x}_0)||_2^2]\}, \tag{4}$$

where $\lambda(i)$ is a non-negative weighting function. When $\mathbf{s}_\theta(\mathbf{x}_i, i)$ is a reasonable predictor of the score function, sampling can be done using, for instance, a reverse diffusion sampler Song et al. [2020]:

$$\mathbf{x}_{i-1} = \mathbf{x}_i - (\sqrt{1-\beta_{i+1}} - 1)\mathbf{x}_i + \beta_{i+1}\mathbf{s}_\theta(\mathbf{x}_i, i) + \sqrt{\beta_{i+1}}\mathbf{z}_i. \tag{5}$$

Another popular sampling method is ancestral sampling [Ho et al., 2020], which is a different discretization of the same reverse-time stochastic differential equation (SDE) [Song et al., 2020]. Since the diffusion sampler (5) can be derived in a conceptually simple manner for an arbitrary SDE, we extensively use it in this paper.

3 Blur diffusion

Coarse-to-fine generation in image synthesis is a successful strategy for both GANs [Karras et al., 2017, 2019] and diffusion models [Ho et al., 2022a, Dhariwal and Nichol, 2021, Nichol and Dhariwal, 2021, Nichol et al., 2021, Ramesh et al., 2022]. The most intuitive way to enable the strategy without separate stages is to define a gradual blurring forward process and reverse it. This can be seen as diffusion in a rotated coordinate system with different velocities for each component of the vector. We first introduce a generalized diffusion process (Sec. 3.1) and propose the blur diffusion as a special case of it (Sec. 3.2).

3.1 Generalized diffusion

A standard diffusion process is defined in the image space directly, assuming the independence between each pixels [Ho et al., 2020]. Our aim is to generalize this process in a rotated coordinate system. For this, we define an orthogonal matrix $\mathbf{U}$, and subsequently some vector rotated by the matrix as $\bar{\mathbf{x}} := \mathbf{U}^T\mathbf{x}$. With slight abuse of notation, we define the fractional powers of a positive semi-definite matrix $\mathbf{P}^p$ as taking the powers of each eigenvalue, i.e, $\mathbf{P}^p = (\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T)^p = \mathbf{U}\mathbf{\Lambda}'\mathbf{U}^T$, where $[\mathbf{\Lambda}']_{ii} = [\mathbf{\Lambda}]_{ii}^p$.

Then, we define a generalized forward diffusion process with the following Markov chain:

$$q(\bar{\mathbf{x}}_i|\bar{\mathbf{x}}_{i-1}) = \mathcal{N}(\bar{\mathbf{x}}_i; (\mathbf{I} - \mathbf{B}_i)^{\frac{1}{2}}\bar{\mathbf{x}}_{i-1}, \mathbf{B}_i\mathbf{I}), \tag{6}$$

where $\mathbf{B}_i$ is a diagonal matrix that defines the noise schedule of the process. Note that (6) is a generalized version of the standard diffusion, as standard diffusion is retrieved when we set $\mathbf{U} = \mathbf{I}$, and $\mathbf{B}_i = \beta_i\mathbf{I}$. In other words, we are introducing more flexibility into the design space of diffusion models by enabling 1) diffusion in the rotated coordinate, where the dependency between pixels can be imposed, 2) diffusion with different velocities for each component of the vector.

Due to the properties of diagonal matrices, we arrive at analytically tractable conditional distribution

$$q(\bar{\mathbf{x}}_i|\bar{\mathbf{x}}_0) = \mathcal{N}(\bar{\mathbf{x}}_i; \bar{\mathbf{A}}_i^{\frac{1}{2}}\bar{\mathbf{x}}_0, (\mathbf{I} - \bar{\mathbf{A}}_i)), \tag{7}$$

where we have defined $\mathbf{A}_i := \mathbf{I} - \mathbf{B}_i$, and $\bar{\mathbf{A}}_i := \prod_{j=1}^{i} \mathbf{A}_j$, analagous to [Ho et al., 2020]. Eq. (7) allows one to directly calculate $\mathbf{x}_i$ using $\mathbf{x}_0$:

$$\mathbf{x}_i = \mathbf{U}\bar{\mathbf{A}}_i^{\frac{1}{2}}\mathbf{U}^T\mathbf{x}_0 + \mathbf{U}(\mathbf{I} - \bar{\mathbf{A}}_i)^{\frac{1}{2}}\epsilon \tag{8}$$

Tractability of Eq. (8) in turn means that we can efficiently train these models with denoising score matching [Vincent, 2011] as in prior studies.

3.2 Blur diffusion

While the choice of rotation matrix $\mathbf{U}$ and the noise schedule $\mathbf{B}_i$ are flexible, here we propose an especially effective choice that can be characterized as blurring diffusion for the forward process. For simplicity and ease of computation, we utilize Gaussian blur with symmetric kernels that are separable, with a pre-defined variance of σ^2. Since Gaussian blur is a linear operation, it can be approximated as a matrix multiplication using a circular symmetric matrix $\mathbf{W}$. With some monotonically increasing function $f(i)$ that determines a blur schedule and $\mathbf{W}_i = \mathbf{W}^{f(i)}$, we define a blurring diffusion process as follows:

$$q(\mathbf{x}_i|\mathbf{x}_{i-1}) = \mathcal{N}(\mathbf{x}_i; \sqrt{1-\beta_i}\mathbf{W}_i\mathbf{x}_{i-1}, \mathbf{C}_i), \tag{9}$$

where we set $\mathbf{C}_i = \mathbf{I} - (1-\beta_i)\mathbf{W}_i^2$ to ensure the process preserves unit variance. Eq. (9) can also be written as follows:

$$\mathbf{x}_i = \mathbf{x}_{i-1} - \mathbf{H}(\mathbf{x}_{i-1}, i-1) + \mathbf{C}_i^{\frac{1}{2}}\mathbf{z}_i, \tag{10}$$

where $\mathbf{H}(\mathbf{x}_i, i) = \mathbf{x}_i - \sqrt{1-\beta_{i+1}}\mathbf{W}_{i+1}\mathbf{x}_i$ is an unnormalized Gaussian high-pass filter. Unlike Eq. (2), where the signal is attenuated holistically, our forward process destroys high frequencies much faster. In order to match the definition of the generalized diffusion, we propose to factor the symmetric matrix $\mathbf{W}$ by eigenvalue-decomposition $\mathbf{W} = \tilde{\mathbf{U}}\mathbf{D}\tilde{\mathbf{U}}^T$ and subsequently $\mathbf{W}_i = \tilde{\mathbf{U}}\mathbf{D}^{f(i)}\tilde{\mathbf{U}}^T$. We employ the memory-efficicent eigen-decompisition by Kawar et al. [2022] (see Appendix D of DDRM). This leads us to the following proposition

Proposition 1. *Let* $\mathbf{B}_i = \mathbf{I} - (1-\beta_i)\mathbf{D}^{2f(i)}$ *and* $\mathbf{U} = \tilde{\mathbf{U}}$. *Then,* (9) *is equivalent to* (6).

Proof. With Eq. (6), $\bar{\mathbf{x}}_i$ is represented as follows:

$$\bar{\mathbf{x}}_i = \sqrt{1-\beta_i}\mathbf{D}^{f(i)}\bar{\mathbf{x}}_{i-1} + (\mathbf{I} - (1-\beta_i)\mathbf{D}^{2f(i)})^{\frac{1}{2}}\mathbf{z}_i \tag{11}$$

Using the definition of $\bar{\mathbf{x}}_i$, we have

$$\mathbf{x}_i = \sqrt{1-\beta_i}\tilde{\mathbf{U}}\mathbf{D}^{f(i)}\tilde{\mathbf{U}}^T\mathbf{x}_{i-1} + \tilde{\mathbf{U}}(\mathbf{I} - (1-\beta_i)\mathbf{D}^{2f(i)})^{\frac{1}{2}}\tilde{\mathbf{U}}^T\bar{\mathbf{z}}_i \tag{12}$$

$$= \sqrt{1-\beta_i}\tilde{\mathbf{U}}\mathbf{D}^{f(i)}\tilde{\mathbf{U}}^T\mathbf{x}_{i-1} + (\tilde{\mathbf{U}}(\mathbf{I} - (1-\beta_i)\mathbf{D}^{2f(i)})\tilde{\mathbf{U}}^T)^{\frac{1}{2}}\bar{\mathbf{z}}_i \tag{13}$$

$$= \sqrt{1-\beta_i}\mathbf{W}_i\mathbf{x}_{i-1} + \mathbf{C}_i^{\frac{1}{2}}\bar{\mathbf{z}}_i, \tag{14}$$

where $\bar{\mathbf{z}}_i \sim \mathcal{N}(0, \mathbf{I})$. Note that $\mathbf{C}_i = \mathbf{I} - (1-\beta_i)\tilde{\mathbf{U}}\mathbf{D}^{2f(i)}\tilde{\mathbf{U}}^T = \tilde{\mathbf{U}}(\mathbf{I} - (1-\beta_i)\mathbf{D}^{2f(i)})\tilde{\mathbf{U}}^T$. □

Due to Proposition 1, we can efficiently train the model using the denoising score matching objective:

$$\mathcal{L} = \mathbb{E}_i\{\lambda(i)\mathbb{E}_{q_0(\mathbf{x})}\mathbb{E}_{q(\mathbf{x}_i|\mathbf{x}_0)}[||\mathbf{s}_\theta(\mathbf{x}_i, i) - \underbrace{(-\mathbf{U}(\mathbf{I} - \bar{\mathbf{A}}_i)^{-1}\mathbf{U}^T\epsilon)}_{=\nabla_{\mathbf{x}_i}\log q_i(\mathbf{x}_i|\mathbf{x}_0))}||_2^2]\}. \tag{15}$$

When we parameterize $\mathbf{s}_\theta(\mathbf{x}_i, i)$ as

$$\mathbf{s}_\theta(\mathbf{x}_i, i) = -\mathbf{U}(\mathbf{I} - \bar{\mathbf{A}}_i)^{-1}\mathbf{U}^T\boldsymbol{\epsilon}_\theta(\mathbf{x}_i, i), \tag{16}$$

Eq. (15) is simplified to:

$$\mathbb{E}_i\{\lambda(i)\mathbb{E}_{q_0(\mathbf{x})}\mathbb{E}_{q(\mathbf{x}_i|\mathbf{x}_0)}[||\mathbf{U}(\mathbf{I} - \bar{\mathbf{A}}_i)^{-1}\mathbf{U}^T(\boldsymbol{\epsilon}_\theta(\mathbf{x}_i, i) - \boldsymbol{\epsilon})||_2^2]\}. \tag{17}$$

In practice, we found it beneficial to sample quality to use the following variant of Eq. (17):

$$\mathcal{L}_\epsilon = \mathbb{E}_i\{\lambda(i)\mathbb{E}_{q_0(\mathbf{x})}\mathbb{E}_{q(\mathbf{x}_i|\mathbf{x}_0)}[||\boldsymbol{\epsilon}_\theta(\mathbf{x}_i, i) - \boldsymbol{\epsilon}||_2^2]\}, \tag{18}$$

which resembles a re-weighted VLB [Ho et al., 2020].

Figure 2: Results on LSUN-bedroom 64 $\times$ 64. f_type : log (*left*), f_type : quartic (*right*).

Figure 3: Comparison of generated images with different generation strategies. Left: fine-to-coarse, right: coarse-to-fine.

Reverse deblurring process After training, we can sample images using a reverse diffusion sampler:

$$\mathbf{x}_{i-1} = \mathbf{x}_i + \mathbf{H}(\mathbf{x}_i, i) + \mathbf{U}\mathbf{B}_{i+1}\mathbf{U}^T\mathbf{s}_\theta(\mathbf{x}_i, i) + \mathbf{U}\mathbf{B}_{i+1}^{\frac{1}{2}}\mathbf{U}^T\mathbf{z}_i \tag{19}$$

or equivalently,

$$\mathbf{x}_{i-1} = \mathbf{x}_i + \mathbf{H}(\mathbf{x}_i, i) - \mathbf{U}\mathbf{B}_{i+1}(\mathbf{I} - \bar{\mathbf{A}}_i)\mathbf{U}^T\boldsymbol{\epsilon}_\theta(\mathbf{x}_i, i) + \mathbf{U}\mathbf{B}_{i+1}^{\frac{1}{2}}\mathbf{U}^T\mathbf{z}_i, \tag{20}$$

which is analogous to unsharp masking [Szeliski, 2010] followed by the denoising term to remove amplified noise. Through the process, our model generates an image in a coarse-to-fine manner by progressive deblurring followed by denoising (see Figure 1).

		FID-10K	
$f(N)$	f_type	bedroom	church
0 (w/o blur)	N/A	9.24	6.04
0.6	log	73.23	
0.14	quartic	**7.86**	**5.89**

Table 1: FID-10K results on LSUN bedroom and church-outdoor datasets (64×64). We fix $f(0)$ to 0.

4 Experiments

Experiment details We set $N = 1000$ and sample using N steps for all experiments. All models are trained on a single V100 with a batch size of 16. We train models for 450K and 600K iterations on LSUN bedroom and church datasets, respectively. We set the learning rate to $5e-5$, EMA decay factor to 0.9999, σ to 0.4, and $\lambda(i)$ to 1. For pre-processing, we resize images to 64×64 without cropping. We do not use any dropout in our experiments. We provide detailed model configuration and computational requirements in Appendix.

Comparison with standard diffusion models In our experiments, we thoroughly compare the proposed blur diffusion to standard diffusion models, which are the special case of our model when $f(i) = 0$. Table 1 demonstrates that our model outperforms the standard diffusion model when f_type is quartic with $f(N) = 0.14$. We compute FID using only 10K samples, and this is acceptable as we measure the relative differences within the same framework. Figure 4 shows the several functional forms of blur schedule we used. As shown in Figure 2 and Table 1, increasing the blur strength too early leads to inferior sample qualities: the model fails to generate reliable high frequencies.

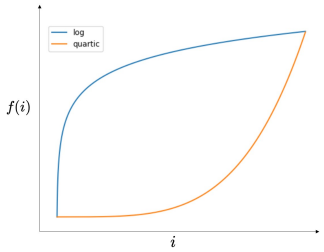

Figure 4: Different functional forms of blur schedule $f(i)$ we experimented with.

Comparison of different sampling strategy We conduct an experiment where we compare our coarse-to-fine approach with fine-to-coarse strategy, for which we replace the diagonal matrix $\mathbf{D}$ with $\mathbf{I} - \mathbf{D}$. As shown in Fig. 3, the fine-to-coarse strategy results in significant artifacts in the generated images, indicating that it is crucial to impose the appropriate inductive bias on diffusion models.

5 Discussion

Perceptual quality While diffusion models can be trained in a perceptual quality-oriented way using hand-crafted weighting functions, our method provides a more explicit way to focus on the coarse pattern of images: to train a score estimator on the blurred images. Moreover, our model emphasizes the coarse patterns during the sampling process as the score function points in the steepest direction to the high-density region of the low-frequencies, especially when i is close to N. Since we did not sweep over blur schedules other than what we reported, it would be a valuable future direction to further find the optimal noise and blur schedule for our model.

Different basis Although we mainly discuss the blur diffusion as a special case, our generalized diffusion is broadly applicable to the arbitrary coordinate depending on the choice of the orthonormal basis $\mathbf{U}$. For instance, one can perform diffusion in different frequency domains of such as Fourier transform or discrete cosine transform. Moreover, our approach is not restricted to the image data and can be used for different data modalities as we provide a general method for imposing the inductive bias on diffusion models.

6 Related Works

Several approaches have been proposed to find a better space for diffusion models. Recently, Vahdat et al. [2021] and Rombach et al. [2022] proposed to train the diffusion model in the learned latent space. Unlike our method, these approaches require the training of an autoencoder and have no control over the learned space, which is necessary to impose the inductive bias for a certain data modality of interest. Jing et al. [2022] utilize the orthogonal projection to destroy the component orthogonal to the data manifold faster. Unlike our work, they focus on reducing the costs for sampling, and the method requires the predetermined time steps in which the projection is performed.

Rissanen et al. [2022] concurrently proposed a deblurring generative process by reversing the heat equation. Indeed, their method is a special case of our generalized diffusion, in which the columns of $\mathbf{U}$ are cosine basis. Although solving heat equation does not involve noise, they empirically found that a small amount of noise (with the variance of 0.1) in the forward process as well as in the generative process is crucial for the sample quality. The noise strength for both processes is chosen by trial and error. In contrast, our approach naturally involves noise as we interpret the proposed diffusion process from the SDE perspective. Therefore, once the noise schedule of the forward process is determined, the reverse-time noise strength is rigorously derived from the forward process, while Rissanen et al. [2022] swept over the reverse-time noise strength δ. Finally, their novel iterative method does not show any improvements in performance yet, while our method demonstrates the improved performance as a generalization of standard diffusion models.

7 Conclusion

In this paper, we generalize the previous diffusion models and provide an effective way to impose the inductive bias on diffusion models. We further propose the blur diffusion as a special case. Blur diffusion generates images in a coarse-to-fine manner by progressive deblurring followed by denoising. Experiments show that our model can synthesize more perceptually compelling samples than previous methods. We look forward to scaling up and applying the model to various applications.

References

B. D. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022a.

J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022b.

B. Jing, G. Corso, R. Berlinghieri, and T. Jaakkola. Subspace diffusion generative models. *arXiv preprint arXiv:2205.01490*, 2022.

T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

B. Kawar, M. Elad, S. Ermon, and J. Song. Denoising diffusion restoration models. *arXiv preprint arXiv:2201.11793*, 2022.

D. P. Kingma, T. Salimans, B. Poole, and J. Ho. Variational diffusion models. *arXiv preprint arXiv:2107.00630*, 2021.

H. Li, Y. Yang, M. Chang, S. Chen, H. Feng, Z. Xu, Q. Li, and Y. Chen. Srdiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 2022.

A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

A. Q. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

S. Rissanen, M. Heinonen, and A. Solin. Generative modelling with inverse heat dissipation. *arXiv e-prints*, pages arXiv–2206, 2022.

R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. 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.

C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021.

J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

R. Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

A. Vahdat, K. Kreis, and J. Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

APPENDIX

A Derivation of reverse diffusion sampler

It was shown in Song et al. [2020] that in the continuous time limit, diffusion models can be viewed as realizations of SDEs. In such view, generative inference can be regarded as discretized solutions to the reverse SDEs. We follow the discretization rules from Song et al. [2020] henceforth to derive our reverse diffusion sampler. Specifically, for a vector-valued function $\mathbf{f}_i(\cdot)$ and matrix $\mathbf{G}_i$, consider the following stochastic difference equation

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{f}_i(\mathbf{x}_i, i) + \mathbf{G}_i \mathbf{z}_i, \tag{21}$$

where $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. With $\bar{\mathbf{f}}(\mathbf{x}_i, i) = N \cdot \mathbf{f}_i(\mathbf{x}_i, i)$ and $\bar{\mathbf{G}}_i = \sqrt{N} \cdot \mathbf{G}_i$, Eq. (21) can be written as

$$\mathbf{x}_{i+1} - \mathbf{x}_i = \bar{\mathbf{f}}(\mathbf{x}_i, i)\Delta t + \bar{\mathbf{G}}_i \sqrt{\Delta t}\mathbf{z}_i, \tag{22}$$

where $\Delta t = \frac{1}{N}$. Let $\mathbf{x}(t) = \mathbf{x}_i, \mathbf{G}(t) = \bar{\mathbf{G}}_i$, and $\mathbf{f}(\mathbf{x}(t), t) = \bar{\mathbf{f}}(\mathbf{x}_i, i)$ for $t = \frac{1}{N}$. In the limit of $N \to \infty$, $\{\mathbf{x}_i\}_i$ becomes a continuous-time stochastic process $\mathbf{x}(t)$ governed by following SDE:

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}(t), t)dt + \mathbf{G}(t)d\bar{\mathbf{w}}. \tag{23}$$

Anderson's theorem [Anderson, 1982] leads us to the following reverse-time SDE of Eq. (23):

$$d\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - \mathbf{G}(t)\mathbf{G}(t)^T \nabla_{\mathbf{x}} \log p_t(\mathbf{x})]dt + \mathbf{G}(t)d\mathbf{w}. \tag{24}$$

For sampling, we further discretize Eq. (24) as follows:

$$\mathbf{x}_{i-1} = \mathbf{x}_i - \mathbf{f}_i(\mathbf{x}_i, i) + \mathbf{G}_i \mathbf{G}_i^T \mathbf{s}_\theta(\mathbf{x}_i, i) + \mathbf{G}_i \mathbf{z}_i, \tag{25}$$

and this is called the reverse diffusion sampler. For blur diffusion, we set $\mathbf{f}_i(\mathbf{x}_i, i) = -\mathbf{H}(\mathbf{x}_i, i)$ and $\mathbf{G}_i = \mathbf{C}_{i+1}^{\frac{1}{2}} = \tilde{\mathbf{U}}\mathbf{B}_{i+1}^{\frac{1}{2}}\tilde{\mathbf{U}}^T$ from Eq. (10), and thus we have:

$$\mathbf{x}_{i-1} = \mathbf{x}_i + \mathbf{H}(\mathbf{x}_i, i) + \tilde{\mathbf{U}}\mathbf{B}_{i+1}\tilde{\mathbf{U}}^T \mathbf{s}_\theta(\mathbf{x}_i, i) + \tilde{\mathbf{U}}\mathbf{B}_{i+1}^{\frac{1}{2}}\tilde{\mathbf{U}}^T \mathbf{z}_i. \tag{26}$$

B Architecture configuration

Our architecture configuration is as follows:

- Diffusion steps: 1000
- Noise schedule: linear
- Model size: 121M
- Channels: 128
- Depth: 3
- Channels multiple: 1,2,3,4
- Heads: 4
- Attention resolution: 4,8
- BigGAN up/downsampling: False
- Dropout: 0
- Batch size: 16
- Iterations: 450K for LSUN bedroom, 600K for LSUN church
- Learning rate: 1e-5

All models are trained on a single V100.