Restoration based Generative Models

Jaemoo Choi [*1] **Yesom Park** [*1] **Myungjoo Kang** [1]

Abstract

Denoising diffusion models (DDMs) have recently attracted increasing attention by showing impressive synthesis quality. DDMs are built on a diffusion process that pushes data to the noise distribution and the models learn to denoise. In this paper, we establish the interpretation of DDMs in terms of image restoration (IR). Integrating IR literature allows us to use an alternative objective and diverse forward processes, not confining to the diffusion process. By imposing prior knowledge on the loss function grounded on MAP-based estimation, we eliminate the need for the expensive sampling of DDMs. Also, we propose a multi-scale training, which improves the performance compared to the diffusion process, by taking advantage of the flexibility of the forward process. Experimental results demonstrate that our model improves the quality and efficiency of both training and inference. Furthermore, we show the applicability of our model to inverse problems. We believe that our framework paves the way for designing a new type of flexible general generative model. The code is available at `https://github.com/Jae-Moo/RGM/`.

1. Introduction

Generative modeling is a prolific machine learning task that the models learn to describe how a dataset is distributed and generate new samples from the distribution. The most widely used generative models primarily differ in their choice of bridging the data distribution to a tractable latent distribution (Goodfellow et al., 2020; Kingma & Welling, 2014; Rezende et al., 2014; Rezende & Mohamed, 2015; Sohl-Dickstein et al., 2015; Chen et al., 2022). In recent years, denoising diffusion models (DDMs) (Ho et al., 2020; Song & Ermon, 2019; Song et al., 2021b; Dockhorn et al., 2022a) have drawn considerable attention by demonstrating remarkable results. DDMs rely on a forward diffusion process that progressively transforms the data into Gaussian noise, and they learn to reverse the noising process. Albeit their enormous successes, their gradual denoising generative process gives rise to low inference efficiency. To pull latent variables back to the data distribution, the denoising process often requires hundreds or even thousands of network evaluations to sample a single instance. Many follow-up studies consider enhancing inference speed (Song et al., 2021a; Tachibana et al., 2021; Lu et al., 2022) or grafting with other generative models (Xiao et al., 2021a; Vahdat et al., 2021; Zhang & Chen, 2021; Pandey et al., 2022).

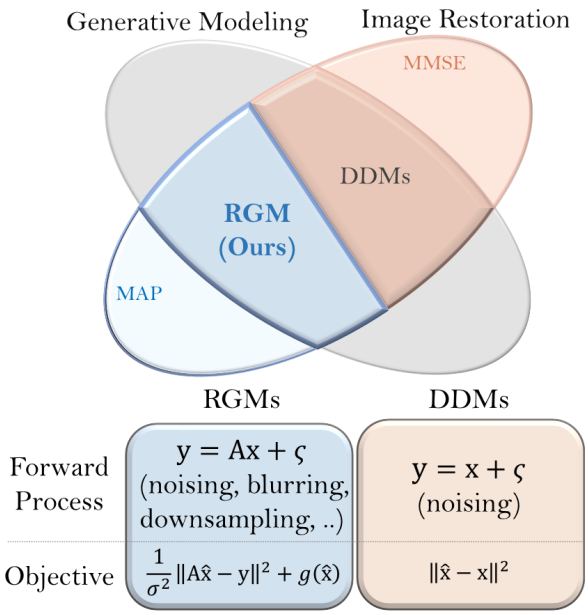

Figure 1: Conceptual comparison of RGMs and DDMs. Here, $\mathbf{x}$ is the original data, $\mathbf{y}$ is the degradation of $\mathbf{x}$ with noise $\xi \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, and $\hat{\mathbf{x}}$ is the reconstruction of $\mathbf{y}$.

In this study, we focus on a different perspective. We interpret the DDMs through the lens of image restoration (IR), which is a family of inverse problems for recovering the original images from corrupted ones (Castleman, 1996; Gunturk & Li, 2018). The corruption arises in various forms, including noising (Rudin et al., 1992; Buades et al., 2005) and downsampling (Farsiu et al., 2004). IR has been a long-standing problem because of its high practical value in various applications (Besag et al., 1991; Banham &

[*]Equal contribution [1]Department of Mathematical Sciences, Seoul National University, Seoul, South Korea. Correspondence to: Myungjoo Kang <mkang@snu.ac.kr>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

Katsaggelos, 1997; Lehtinen et al., 2018; Ma et al., 2011). From the IR point of view, DDMs can be considered as IR models based on minimum mean square error (MMSE) estimation (Zervakis & Venetsanopoulos, 1991; Laumont et al., 2022), focusing only on the denoising task. Mathematically, IR is an ill-posed inverse problem in the sense that it does not admit a unique solution (Hadamard, 1902) and hence, leads to instability in reconstruction. Owing to the ill-posedness of IR, MMSE produces impertinent results. DDMs alleviate this problem by leveraging costly stochastic sampling, which has been regarded as an indispensable tool in the literature on DDMs.

Inspired by this observation, we propose a new flexible family of generative models that we refer to *restoration-based generative models (RGMs)*. We adopt an alternative objective; a maximum a posteriori (MAP) based regularization (Hunt, 1977; Trussell, 1980), which is predominantly used in IR. This approach detours the ill-posedness by regularizing the data fidelity loss by prior knowledge rather than doing costly iterative sampling. Prior knowledge can be utilized in a variety of ways. Moreover, we also have the freedom to design the degradation process. RGMs with variability of these two have several benefits:

Implicit Prior knowledge Many hand-crafted regularization schemes (Tikhonov, 1963; Donoho, 1995; Baraniuk, 2007) encourage solutions to satisfy certain properties, such as smoothness and sparsity. However, for the purpose of density estimation, we parametrize the prior term to learn the statistical distance, e.g., Kullback–Leibler divergence or Wasserstein distance. We also introduce a random auxiliary variable to further ease the ill-posedness. Our MAP-based estimation allows a much smaller computational cost, retaining the density estimating capability of DDMs.

Various Forward process DDMs are buried in a Gaussian noising process. On the other hand, fluidity in the forward process of RGMs improves model performance because the behavior of generative models is significantly affected by how the data distribution is transformed into a simple distribution. As one instantiation, we design a degradation process that progressively reduces the dimension by block averaging the image, which improves performance.

Our comprehensive empirical studies on image generation and inverse problems demonstrate that RGMs generate samples rivaling the quality of DDMs with several orders of magnitude faster inference. In particular, our model achieves FID 2.47 on CIFAR10, with only seven network function evaluations. Furthermore, through rigorous experiments with various prior terms and degradation, we validate that our RGM framework is well-structured that opens the way for designing more efficient and flexible generative models.

2. Background

Image Restoration A common inverse problem arising in image processing, including denoising and inpainting, is the estimation of an image $\mathbf{x}$ given a corrupted image

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \xi, \tag{1}$$

where $\mathbf{A}$ is a matrix that models the degradation process, and $\xi \sim \mathcal{N}(0, \mathbf{\Sigma})$ is an additive noise. A family of such problems is known as image restoration (IR). A popular approach is the maximum-a-posteriori (MAP) estimation

$$\operatorname{argmax}_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) = \log p(\mathbf{y} \mid \mathbf{x}) + \log p(\mathbf{x}). \tag{2}$$

However, since the explicit density function $\log p(\mathbf{x})$ is intractable, they replace the objective (2) with

$$\operatorname{argmin}_{\mathbf{x}} f(\mathbf{y}, \mathbf{x}) + g(\mathbf{x}), \tag{3}$$

where $f(\mathbf{x}, \mathbf{y}) = -\log p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{2} \left\| \left(\mathbf{\Sigma}^{\dagger}\right)^{\frac{1}{2}} (\mathbf{A}\mathbf{x} - \mathbf{y}) \right\|_2^2$ is the data fidelity term with the pseudoinverse (Moore, 1920) $\mathbf{\Sigma}^{\dagger}$ and g is a regularization term (or prior knowledge) that represents prior or constraints on the solution. Since (3) originated from the MAP objective, it is also called the MAP-based approach. Prior knowledge can be imposed in a variety of ways and the choice is crucial because the quality of the restoration varies according to different prior. Moreover, the ill-posedness nature of the inverse problem (1), that is non-uniqueness of the solution, necessitates the use of regularization. By imposing certain prior information about the desirable recovery, the prior knowledge g relieves the ill-posedness. Therefore, many researchers have been devoted to designing a proper prior g (Rudin et al., 1992; Mallat, 1999; Lunz et al., 2018).

Denoising Generative Models Denoising diffusion models (DDMs) (Ho et al., 2020; Song et al., 2021b) have recently emerged as the forefront of image synthesis research. Starting from the image distribution, they gradually corrupt the image $\mathbf{x}_0 \sim p_{\text{data}}$ into Gaussian noise over time through a forward diffusion process with a given noise schedule σ_t:

$$q^{(t)}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_0, \sigma_t^2 \mathbf{I}\right),$$

also known as VESDE (Song et al., 2021b). Another linear diffusion process named VPSDE is also leveraged, but since these two are known to be exchangeable with each other (Kim et al., 2022), the paper focuses on VESDE. DDMs pose the data generation as an iterative denoising procedure by learning the reverse of the forward process. As they are modeled with conditional Gaussian distributions, evidence lower bound (ELBO) (Sohl-Dickstein et al., 2015) could be simplified to the following objective with a weight $\lambda(t) \geq 0$ (Ho et al., 2020; Song et al., 2021b):

$$\Sigma_{t=0}^{T} \mathbb{E}_{\mathbf{x}_0 \sim p_{\text{data}}, \mathbf{x}_t \sim q_{\sigma_t}(\mathbf{x}_t \mid \mathbf{x}_0)} \left[\lambda(t) \left\| G_\theta(\mathbf{x}_t, t) - \mathbf{x}_0 \right\|_2^2 \right], \tag{4}$$

where G_θ is a neural network parametrized by θ.

For each forward step t, (4) is the minimum mean square error (MMSE) objective. MMSE loss is simple and straightforward to train, however, the solution is only optimized to ensure accordance with the degradation process because it only contains the recovery term. Consequently, MMSE is affected by ill-posedness. To be precise, when σ_t is large, (1) becomes highly ill-posed and possesses many solutions for a given observation. In this case, the MMSE solution averages all these candidate solutions, resulting in an atypical reconstruction. Recent works (Laumont et al., 2022; Kawar et al., 2021) have endeavored to resolve this problem by stochastic sampling, however, they suffer from notoriously low efficiency as they roll out thousands of trajectories. In a similar manner, DDMs utilize a sampling scheme that often requires hundreds to thousands of steps. In summary, there are two limitations of DDMs from the IR perspective:

1. The degradation process is restricted to Gaussian noising.
2. The inference efficiency is intrinsically low due to the MMSE estimator.

3. Method

3.1. MAP-based Estimation for Generation

As alluded in Section 2, DDMs can be regarded as MMSE-grounded IR models, specialized in denoising. This observation brings us a new perspective on the design of a family of flexible generative models. As an alternative to MMSE, we propose a new generative model based on (3):

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}},\mathbf{y}\sim\mathcal{N}(\mathbf{x},\sigma^2\mathbf{I})}\left[\frac{1}{2\sigma^2}\|G_\theta(\mathbf{y})-\mathbf{y}\|_2^2+\lambda g(G_\theta(\mathbf{y}))\right], \tag{5}$$

where the first term measures the data fidelity and the second term delivers the prior knowledge of the data distribution. It has been adopted as a standard approach for high-dimensional imaging problems and is known to be more relevant than MMSE in many applications (Saha et al., 2009; Bigdeli et al., 2019; Chen, 2016). By leveraging prior information on the solution, MAP-based approaches alleviate the ill-posedness of the inverse problem (1), without the use of costly sampling. Therefore, carefully crafting the relevant prior term is crucial. We now show how one can execute an appropriate prior term for density estimation while alleviating the ill-posedness.

Alleviation of ill-posedness Unlike the generic denoising task, it is necessary to bridge the image to the Gaussian noise to learn the data distribution. As the noise level increases, a single distorted observation has several solutions, which indicates that the ill-posedness deepens. Therefore, our generator G_θ should be able to recover diverse restorations from a degraded image to express the data distribution more abundantly. Since it is difficult for the regularization term to remedy all these problems on its own, we further offload the ill-posedness by introducing a random auxiliary variable $\mathbf{z}\sim\mathcal{N}(\mathbf{z}\mid\mathbf{0},\mathbf{I})$. In other words, we use the stochastic variable $\mathbf{z}$ as an input of G_θ. As $G_\theta(\mathbf{y},\mathbf{z})$ generates various restores for different $\mathbf{z}$, it is more amenable to faithfully recovering the data distribution.

Implicit Prior Knowledge For density estimation, the knowledge about the data distribution should be properly encoded in the prior term g of (5). However, since the explicit density of the data is inaccessible, we parameterize g to learn the prior. For each forward step, our new objective for the generator G_θ in conjunction with the $\mathbf{z}$ is given by:

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}},\mathbf{y}\sim\mathcal{N}(\mathbf{x},\sigma^2\mathbf{I}),\mathbf{z}\sim\mathcal{N}(0,I)}\left[\frac{1}{2\sigma^2}\|G_\theta(\mathbf{y},\mathbf{z})-\mathbf{y}\|_2^2 + \lambda g_\phi(G_\theta(\mathbf{y},\mathbf{z}))\right], \tag{6}$$

where g_ϕ is a learnable prior term parameterized by ϕ. For example, we can learn an implicit representation of the data by adopting $g_\phi(\mathbf{x})=\log\left(1-D_\phi\left(\mathbf{x}\right)\right)-\log D_\phi\left(\mathbf{x}\right)$ where D_ϕ is a discriminator trained coupled with G_θ. As D_ϕ gets close to the optimal, i.e., $D_\phi(x)=\frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_\theta(x)}$, the expectation of g_ϕ approaches to Kullback–Leibler divergence (KLD) $D_{\text{KL}}(p_\theta||p_{\text{data}})$, which leads the loss (6) with $\lambda=1$ to agree with the following objective:

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}},\mathbf{y}\sim\mathcal{N}(\mathbf{x},\sigma^2\mathbf{I})}\left[D_{KL}(p_\theta(\mathbf{x}|\mathbf{y})||p(\mathbf{x}|\mathbf{y}))\right]+\mathcal{H}(p_\theta), \tag{7}$$

where $\mathcal{H}$ denotes an entropy and p_θ the distribution generated by G_θ. The overall training procedure combined with all $\sigma\in\{\sigma_k\}_{k=1}^T$ is provided in Appendix B.2.

Remark 3.1. Contrary to conventional IR literature whose prior term is pre-defined, our approach (6) tries to learn the prior term by coordinating with G_θ. This end-to-end training allows our MAP-inspired scheme to deliver more promising performance. Moreover, it is worth noting that our framework has the wide freedom in the choice of g_ϕ without being tied to the discriminative learning for KLD exemplified above. Consequently, we further design the prior term by mounting maximum mean discrepancy (MMD) (Dziugaite et al., 2015) and distributed sliced Wasserstein distance (DSWD) (Nguyen et al., 2020) in Section 4.

Small Denoising Steps A major downside of DDMs is their sampling inefficiency, which often requires hundreds to thousands of denoising steps to obtain a single image. By adopting regularization term g, our approach provides an avenue to offload the time-consuming sampling and enables significantly small denoising steps. For small degradation, we can obtain a restored image in one shot. But, as our restoration starts from the Gaussian noise, the data distribution is not completely estimated. Therefore, we perform the

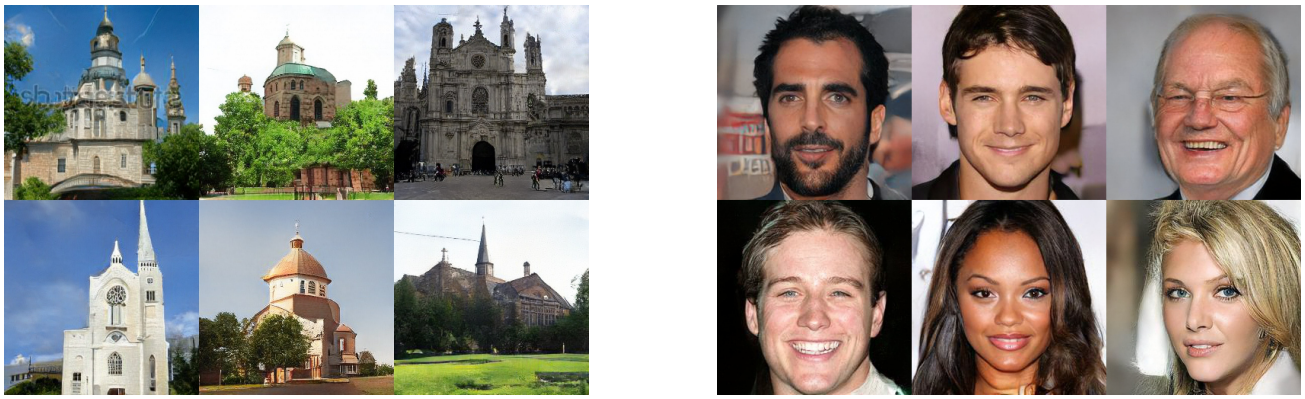

Figure 2: Generated samples on LSUN Church (left) and CelebA-HQ (right).

generation iteratively. In our experiments on CIFAR10, we generate a high-quality sample in four denoising steps.

3.2. Extension to General Restoration

In Section 3.1, we proposed a denoising generative model based on MAP objective. However, from the IR perspective, it is not necessary to restrict the forward process to Gaussian noising ($\mathbf{A} = \mathbf{I}$) and it can be generalized to any family of degradation matrices $\mathbf{A}$ and noise factors $\mathbf{\Sigma}$ in (1). Utilizing the general forward process, we can learn the generative model by generalizing the loss function (6) as follows:

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}},\mathbf{y}\sim\mathcal{N}(\mathbf{A}\mathbf{x},\mathbf{\Sigma}),\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\Big[\lambda g_\phi(G_\theta(\mathbf{y},\mathbf{z})) + \frac{1}{2}\left\|\left(\mathbf{\Sigma}^\dagger\right)^{\frac{1}{2}}\left(\mathbf{A}\cdot G_\theta(\mathbf{y},\mathbf{z})-\mathbf{y}\right)\right\|_2^2\Big]. \tag{8}$$

Therefore, RGM has an flexible structure that can permeate any forward process, and aids in designing a new generative model. Here, we propose a new model established upon super-resolution (SR).

Multi-scale RGM Most DDMs maintain the image size during the diffusion process by adding noise to individual pixels. Consequently, they are very inefficient because they require a latent as much as dimension of pixel space that is much larger than the submanifold of the image space. Motivated by this, we take $\mathbf{A}$ as a block averaging filter that averages out 2×2 pixel values. Halving the image size at each coarsening step allows us a more expressive generative model with a lower-dimensional latent distribution. Moreover, multi-scale training has proven to be an effective strategy for synthesizing large scale images (Denton et al., 2015; Karras et al., 2017b; Reed et al., 2017). Therefore, our model produces strikingly realistic images by progressively extracting spatial information.

4. Experiments

This section evaluates the performance of the proposed RGMs on synthetic and several benchmark datasets. We also analyze our model through extensive ablation studies. Furthermore, we show the capability of RGMs for solving inverse problems. We parametrize G_θ based on the UNet-like structure (Ronneberger et al., 2015) which was successfully used in NCSN++ (Xiao et al., 2021a). The internal details of the implementation can be found in Appendix B.

Setup Our RGMs have a free hand in designing the forward process (i.e. data fidelity) and the prior term. To verify the pliability of RGMs, we implement RGMs with diverse forward processes and regularization terms:

- We consider three prior knowledge by leveraging KLD, MMD, and DSWD, where each stands for the measurement of the difference between two distributions. KLD measures how much two distributions diverge from each other entropically as introduced in Section 3.1. Using a kernel trick, MMD measures the mean squared difference of the statistics of two sets of samples. DSWD estimates the difference by calculating the sliced Wasserstein distance for two distributions for multiple projection vectors. With the Gaussian noise forward process, we call these models *RGM-KLD-D*, *RGM-MMD-D*, and *RGM-DSWD-D*, respectively. Here, the term "D" stands for "denoising". See Appendix B.2 for a detailed explanation.

- Additional to the Gaussian noise forward process, which is primarily used in DDM, we also carry out an RGM-KLD with the multi-scale forward process introduced in Section 3.2. In this case, the image is corrupted by a downsampling filter together with additive noise. Therefore, the model (termed by *RGM-KLD-SR (naive)*) is demanded to conduct upsampling and denoising at the same time. To ease the task, we separate the downsampling and noising processes and perform them alternatively. We call a model using this separated schedule *RGM-KLD-SR*. (See Appendix B.1 for details).

4.1. 2D Toy Example

We first employ a two-dimensional example to validate the effectiveness and flexibility of our framework. We adopt

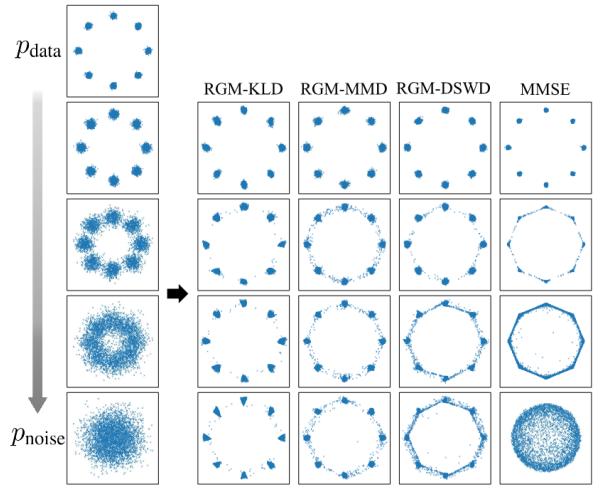

Figure 3: Comparison of recovering density by MMSE versus three RGMs with different priors. All three RGMs are much more efficient than the MMSE approach.

a mixture of Gaussian with eight components (Grathwohl et al., 2018) as a target distribution. Our results are depicted in Figure 3. As illustrated in the first column, we diffuse the data distribution through four different noise levels. From left to right, each of the columns represents the learned distribution of our RGMs with the regularization term parameterized by KLD, MMD, DSWD, and lastly MMSE estimation.

Effectiveness of MAP-based approach Figure 3 shows the benefits of our methodology over the MMSE approach. First, the rightmost column shows the failure of MMSE, where the modes of the distribution are connected and then missed. This tendency exacerbates as the noise level increases. Since the MMSE fails to reconstruct the data distribution even with a small rise in the noise level, the MMSE does not yield a satisfactory generative model with a small number of diffusion steps. Consequently, MMSE approaches, such as DDMs, require a large number of steps to stably recover the data distribution. On the other hand, by adding the prior knowledge our RGMs generate samples from the multimodal distribution significantly better, which allow distribution recovery with a much smaller number of forward processes than the MMSE approach. This demonstrates the effectiveness of using the prior term g.

Flexibility of the prior term Our RGMs have the freedom to parametrize the prior term g of (6). To demonstrate that the RGM framework universally works for variously parametrized prior terms, we manifoldly design the prior term by KLD, MMD, and DSWD. The results depicted in Figure 3 validate that RGMs parametrized in three different ways show consistent performance, where they are all more efficient than the MMSE estimator. In particular, MMD measures the distance between two distributions based on a pre-defined kernel, and hence g is fixed rather than learned.

Table 1: Results on unconditional generation of CIFAR10.

Class	Model	FID (↓)	IS (↑)	NFE (↓)
RGM	RGM−DSWD−D	3.11	9.08	4
	RGM-KLD-D	3.04	9.14	4
	RGM-KLD-SR	2.47	9.68	7
DDM	NCSN (Song & Ermon, 2019)	25.3	8.87	1000
	DDPM (Ho et al., 2020)	3.21	9.46	1000
	Score SDE (VE) (Song et al., 2021b)	2.20	9.89	2000
	Score SDE (VP) (Song et al., 2021b)	2.41	9.68	2000
	Probability Flow (VP) (Song et al., 2021b)	3.08	9.83	140
	DDIM (50 steps) (Song et al., 2021a)	4.67	8.78	50
	Recovery EBM (Gao et al., 2021)	9.58	8.30	180
	LSGM (Vahdat et al., 2021)	2.10	9.87	147
	FastDDPM ($T = 50$) (Kong & Ping, 2021)	3.41	8.98	50
	VDM (Kingma et al., 2021)	4.00	-	1000
	UDM (Kim et al., 2021)	2.33	10.1	2000
	GGF (Jolicoeur-Martineau et al., 2021b)	2.44	-	1000
	Subspace Diffusion (Jing et al., 2022)	2.17	9.94	$\geq$ 1000
	CLD (Dockhorn et al., 2022a)	2.25	-	2000
	DDGAN (Xiao et al., 2021a)	3.75	9.63	4
	DEIS (Zhang & Chen, 2022)	3.37	9.74	15
	StyleGAN2+ES-DDPM (Lyu et al., 2022)	5.52	-	101
	DPM-Solver-3 (Lu et al., 2022)	2.70	-	30
	GENIE (Dockhorn et al., 2022b)	3.94	-	20
GAN	SNGAN+DGflow (Ansari et al., 2020)	9.62	9.35	25
	AutoGAN (Gong et al., 2019)	12.4	8.60	1
	TransGAN (Jiang et al., 2021)	9.26	9.02	1
	StyleGAN2 w/o ADA (Karras et al., 2020)	8.32	9.18	1
	StyleGAN2 w/ ADA (Karras et al., 2020)	2.92	9.83	1
Others	NVAE (Vahdat & Kautz, 2020)	23.5	7.18	1
	Glow (Kingma & Dhariwal, 2018)	48.9	3.92	1
	PixelCNN (Van Oord et al., 2016)	65.9	4.60	1024
	VAEBM (Xiao et al., 2020)	12.2	8.43	16

Despite this simple structure, the results confirm that our RGM with MMD is more efficient than MMSE.

4.2. Image Generation

We compare the performance of our RGMs with several existing baselines. We use Fréchet Inception Distance (FID) and Inception Score (IS) as the evaluation metrics. We also report the number of network function evaluations (NFEs). For DDMs and RGMs, NFE value and real inference time are proportional. Following (Song et al., 2021b; Dockhorn et al., 2022a), we focus on the widely used CIFAR10 unconditional image generation benchmark (Krizhevsky et al., 2009) and also validate the performance of RGMs on large-scale (256×256) images: CelebA-HQ (Liu et al., 2015) and LSUN Church (Yu et al., 2015). Tables 1 and 2 summarize the quantitative evaluations on CIFAR10 and CelebA-HQ, respectively. The qualitative performance of RGM-KLD-D is depicted in Figures 2 and 4.

Results We can see that our models are comparable to the best existing DDMs on CIFAR10 and achieve the state-of-the-art FID score on CelebA-HQ-256. Although the best denoising models obtain better results than ours on CIFAR10, they use a much larger number of denoising steps (e.g. ScoreSDE with VESDE requires 2000 steps). Notably, our RGM-KLD-SR achieves FID 2.47 and IS 9.68 with

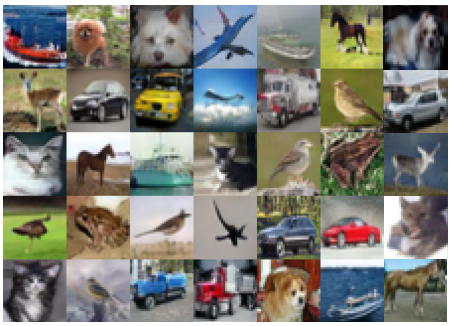

Figure 4: CIFAR10 generated samples.

only seven steps, which is state-of-the-art sampling FID performance when NFE is limited. The overall results confirm that our method immediately eliminates the need for an expensive sampling scheme while still maintaining the density estimating capability of DDMs. Interestingly, RGM-KLD-SR outperforms RGM-KLD-D by a large margin even with far fewer latent variables than RGM-KLD-D. This improved performance may be attributed to the increase in NFE; however, the FID of RGM-KLD-D with $T = 8$ reported in Table 7 confirms that it is not. In addition, RGM with the DSWD prior term retains comparable performance. This verifies that our MAP-inspired objective (6) works universally well, not being tied to how we parametrize the prior term. The overall results indicate that the prior knowledge regularized estimation of RGMs is a promising way of generating high-quality samples in limited steps. More uncurated images can be founded in Appendix C.5.

Table 2: Results on generation of CelebA-HQ-256.

Class	Model	FID (↓)	NFE (↓)
RGM	RGM-KLD-D	**7.15**	4
DDM	Score SDE (VP) (Song et al., 2021b)	7.23	4000
	Probability Flow (Song et al., 2021b)	128.13	335
	LSGM (Vahdat et al., 2021)	7.22	23
	UDM (Kim et al., 2021)	7.16	2000
	DDGAN (Xiao et al., 2021a)	7.64	4
GAN	PGGAN (Karras et al., 2017a)	8.03	1
	Adv. LAE (Pidhorskyi et al., 2020)	19.2	1
	VQ-GAN (Esser et al., 2021)	10.2	1
	DC-AE (Parmar et al., 2021)	15.8	1
VAE	NVAE (Vahdat & Kautz, 2020)	29.7	1
	VAEBM (Xiao et al., 2020)	20.4	1
	NCP-VAE (Aneja et al., 2021)	24.8	1

4.3. Ablation Studies

This section is devoted to validating that the structure of the RGM framework is well-organized, with all parts of our objective, including fidelity term, prior knowledge, auxiliary variable, and regularization parameter, faithfully fulfilling their respective roles. For a fair comparison, we used the same network for all experiments.

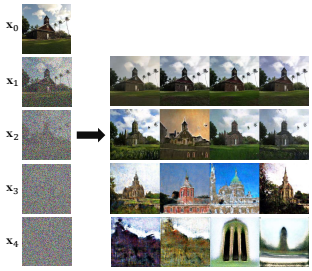

Figure 5: Study on the effect of auxiliary variable $\mathbf{z}$.

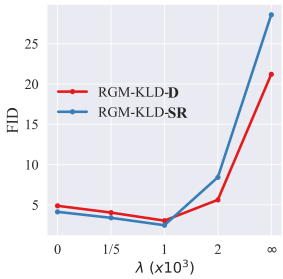

Figure 6: FIDs for different regularization parameter λ.

On the effect of Varying λ We investigate the sensitivity of the regularization parameter λ in (6). Since it controls the relative importance between the fidelity term and the prior term, λ is a trade-off hyperparameter that determines how much regularizes the joint distribution of p_k and p_{k+1}. In Figure 6, we present FID scores measured on CIFAR10 with the same number of degradation steps ($T = 4$) and varying λ. We can see that our models are quite robust with respect to λ. An empirically observed sweet spot of λ is $d/10 \leq \lambda \leq d$ for the image size d, in which FID is no longer improved outside this threshold. For small λ, the models put a lot of effort to recover the degradation, which hinders estimating data distribution. Choosing a large λ also results in performance degeneration.

On the Importance of Fidelity term The results of RGMs trained without the fidelity term also draw our attention. Table 3 shows that the FID scores of both RGM-KLD-D and RGM-DSWD-D degenerate when there is no fidelity term. In particular, looking at RGM-DSWD-D, the performance without fidelity term is inferior to the vanilla DSWD model despite using multiple timesteps. This demonstrates that the performance improvement of our model is not solely due to the power of the existing generative models we used to design the prior knowledge. On the other hand, including the fidelity term significantly improves performance. In particular, RGM-DSWD-D achieves more than two times performance improvement over the vanilla DSWD. The ablation studies we discuss here confirm that RGMs owe the performance improvement to the fidelity term, not simply because we borrow the expressivity from the regularization term. We further observe that with the help of the fidelity term, our model enhances the mode-collapsing resiliency of GAN (See Figure 18). Overall results validate that the performance of RGMs owes to both fidelity and prior term, and the reliable regularization parameter should be determined to balance these two terms.

On the role of $\mathbf{z}$ We include experimental results on LSUN church, which demonstrate how the auxiliary variable $\mathbf{z}$ alleviates the ill-posedness of the inverse problem. By noising the upper-left image $\mathbf{x}_0$, we obtain the forward trajectory $\{\mathbf{x}_k\}_{k=1}^4$. The figures on the right are restored

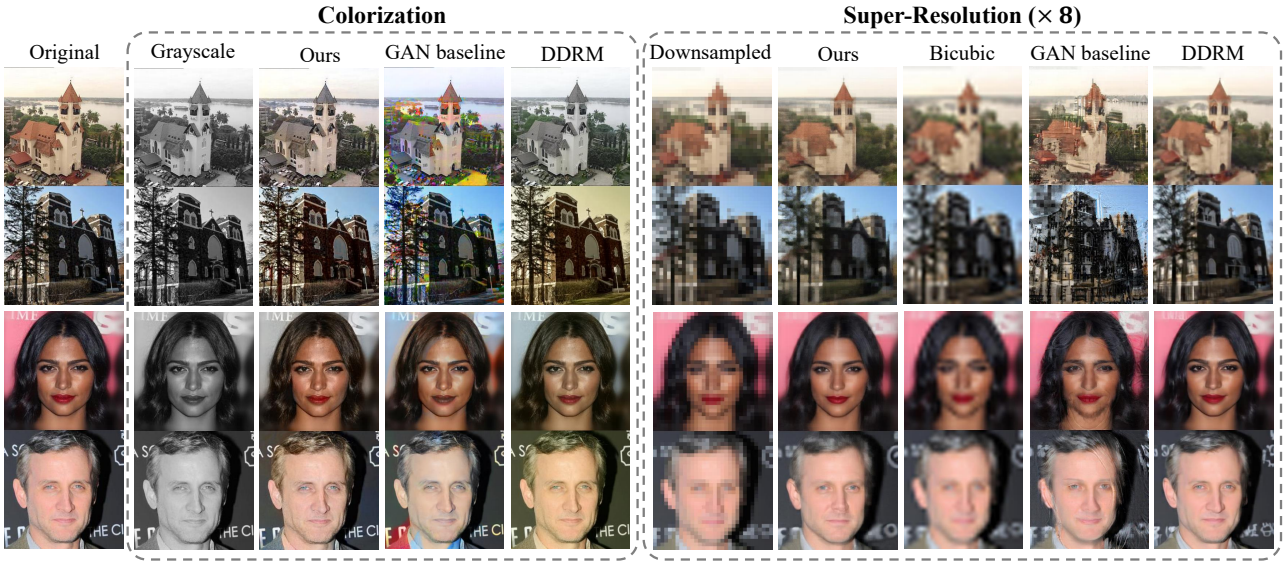

Figure 7: Colorization (left) and super-resolution (right) results on LSUN and CelebA-HQ datasets.

Table 3: Ablation studies of RGMs on CIFAR10.

Model	Multi-step	Fidelity	$\mathbf{z}$	FID (↓)
RGM-KLD-D	✗	✗	✗	42.8
	✗	✓	✓	14.6
	✓	✗	✓	32.5
	✓	✓	✗	3.87
	✓	✓	✓	3.04
RGM-DSWD-D	✗	✗	✗	7.12
	✓	✗	✓	16.3
	✓	✓	✓	3.14
RGM-KLD-SR (naive)	✓	✓	✓	3.17
RGM-KLD-SR	✓	✓	✓	2.47

images of $\mathbf{x}_k$ by RGM-KLD-D together with four different $\mathbf{z}$. We can see that reconstruction is almost unique when the noise level is small. But, as the noise level increases, a single $\mathbf{x}_k$ has various reconstructions. It is evident that assigning $\mathbf{z}$ helps generate different denoised images from a heavily degraded $\mathbf{x}_k$ through the guidance provided by $\mathbf{z}$. However, one might think that the ill-posedness is detoured by multi-step training using multiple σ_k rather than through $\mathbf{z}$. This claim can be refuted using the result of RGM-KLD-D without $\mathbf{z}$ reported in Table 3. We observe the significant difference in FIDs of RGM-KLD-D with and without $\mathbf{z}$ under the same number of denoising steps, which indicates the effectiveness of $\mathbf{z}$.

On the forward process schedule Since the forward process determines the way of connecting the data and latent distributions, it significantly affects the performance of models. The first important factor is the number of forward steps T, which is directly related to NFE. In Table 3, we ablate the effect of T. When $T = 1$, it may be difficult for the model to directly approximate the data distribution from the Gaussian noise. This is reflected in the poor FID score.

We also study the forward process schedule of the SR model. We can observe that the separation of the same forward process into two steps makes the model easier to learn, and this brings the performance enhancement of RGM-KLD-SR compared to RGM-KLD-SR (naive). From this, we would like to point out that properly designing the forward process can significantly increase performance. We leave the development of more useful and rigorous forward process as a promising future direction.

4.4. Inverse Problems

While our model was originally devised to generate images, we further show the applicability of RGMs to inverse problems. Recently, a promising approach in imaging inverse problems is to leverage a learned denoiser as an alternative to the proximal operator of splitting algorithms (Romano et al., 2017; Hurault et al., 2021). Such methodology is referred to as Plug-and-Play (PnP) algorithms (Venkatakrishnan et al., 2013). In a similar spirit, we utilize the trained RGMs as a modular part of the PnP algorithms to solve various inverse problems. In this section, we testify our RGM-KLD-D for two inverse problems; SR and colorization, by plugging our model into Douglas-Rachford Splitting algorithm (Lions & Mercier, 1979). Details can be found in Appendix B.4.

Results We compare the performance of our model with current-leading models: We compare our model with DDRM (Kawar et al., 2022), which solves inverse problems with a pre-trained DDPM by a posterior sampling scheme. As a GAN baseline, we adopt StyleSwin (Zhang et al., 2022) and reconstruct the image by optimizing over the latent vector (Pan et al., 2021). We also consider bicu-

Table 4: Quantitative comparison of RGM-KLD-D and RGM-KLD-SR on image reconstruction.

Model	Super-Resolution				Denoising					
	($\times 2$)		($\times 4$)		($\sigma = 10/255$)		($\sigma = 20/255$)		($\sigma = 40/255$)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
RGM-KLD-D	26.63	0.88	20.84	0.58	**30.11**	**0.93**	**26.57**	**0.86**	**24.23**	**0.80**
RGM-KLD-SR	**27.42**	**0.90**	**21.14**	**0.59**	29.41	0.92	25.87	0.84	23.53	0.77

bic interpolation as a baseline for super-resolution. We observe that our model is capable of reconstructing faithful and realistic images, as evident in Figure 7. Compared with baselines, our model produces high-quality reconstructions across all the datasets. In particular, our model shows promising performance for colorization. These results show the applicability of RGMs to PnP prior, and this will bring a range of potential applications, including image segmentation, conditional generation, and other imaging inverse problems. Additional quantitative and qualitative results are provided in Appendix C.3.

Comparison of RGM-D and RGM-SR We investigate the effect of the degradation process used during training on the performance of solving inverse problems. We compare the reconstruction performance of RGM-KLD-D and RGM-KLD-SR which are trained on different degradation processes by applying both models to denoising and super-resolution (SR) tasks on CIFAR10. Quantitative results are presented in Table 4. We can see that the RGM-KLD-SR that is trained based on SR actually performs the SR task better. Also, we can observe a similar tendency for denoising. The results confirm that the degradation process used in training actually helps in solving the corresponding inverse problem.

5. Related Work

In recent years, DDMs (Ho et al., 2020; Song & Ermon, 2019; Song et al., 2021b) have emerged as a class of density estimation models, first sparked by (Sohl-Dickstein et al., 2015). They define a sampling process as the reverse of a forward diffusion process that maps data to Gaussian noise by consecutively adding a small portion of the noise to the input data. DDMs are known to faithfully estimate the data distribution and generate high-fidelity samples, however, their major drawback is slow and expensive sampling speed. Many studies have been dedicated to circumventing this downside by developing a fast numerical solver (Jolicoeur-Martineau et al., 2021a; Zhang & Chen, 2022; Tachibana et al., 2021; Liu et al., 2022) or using an alternative noising process such as non-Markovian (Song et al., 2021a), a second-order Langevin dynamics (Dockhorn et al., 2022a), and non-linear diffusion processes (De Bortoli et al., 2021; Chen et al., 2022). Another line of work improves sampling efficiency by incorporating it into other generative models, including GAN (Xiao et al., 2021a; Lyu et al., 2022), and VAE (Vahdat & Kautz, 2020). Xiao et al. (2021a) which enjoys small sampling steps by using GAN is one of our related works. On a side note, all the aforementioned models use the Gaussian noising process as the forward process.

Recently, the literature has begun to replace the additive Gaussian noising process with other transforms. Breaking away from the diffusion process, (Rissanen et al., 2022) proposed a forward blurring process inspired by heat dissipation. They suggest a new generation process, but they specialize in the proposed blurring process and cannot be incompatible with other degradation processes. Possibly the closest study to our work is Cold Diffusion (Bansal et al., 2022) which generalizes the diffusion process to arbitrary image transformations. It seems to use a general transform similar to our models, but Cold Diffusion only uses deterministic degradation processes by entirely removing additive Gaussian noise, which hinders its density estimation performance. Also, they use the MMSE objective, still requiring an array of several forward steps. We include a comparison with these related works in Appendix C.2.

6. Conclusion and Future Work

In this study, we presented a general framework for modeling efficient generative models through the lens of IR. Compared to DDMs whose both forward and reverse processes are fixed to thousands of Gaussian steps, our approach provides more flexible models that eliminate expensive sampling and can enjoy versatile forward processes. We eliminated the usage of slow sampling by taking on the MAP-based approach and incorporating implicit priors. In addition, we propose a multi-scale method as an example of the usability of various forward processes. The experimental results showed that the image quality obtained was on par with the leading DDMs, and we achieved state-or-the-art performance using a limited number of forward steps. We hope that this work provides a broad view of modeling useful generative models.

Our model has two degrees of freedom: One is how to parametrize the prior knowledge, and the other is the choice of the forward process. Designing new prior terms and degradation processes would be an interesting direction for future research. Future work could include the comprehensive design of a convergence-guaranteed PnP algorithm for

application to various inverse problems. We leave these further extensions to future work. Furthermore, notwithstanding the high performance, our methodology lacks theoretical justification. We also leave this as interesting future work.

Acknowledgements

This work was supported by the NRF grant [2012R1A2C3010887] and the MSIT/IITP ([1711117093], [2021-0-00077], [No. 2021-0-01343, Artificial Intelligence Graduate School Program(SNU)]). The author appreciate the financial support provided by the Data-driven Flow Modeling Research Laboratory funded by the Defense Acquisition Program Administration under Grant UD230015SD. We are also grateful to Jaewoong Choi and Changyeon Yoon for reviewing an early draft of this paper and providing thoughtful feedback.

References

Aneja, J., Schwing, A., Kautz, J., and Vahdat, A. A contrastive learning approach for training variational autoencoder priors. *Advances in neural information processing systems*, 34:480–493, 2021.

Ansari, A. F., Ang, M. L., and Soh, H. Refining deep generative models via discriminator gradient flow. *arXiv preprint arXiv:2012.00780*, 2020.

Banham, M. R. and Katsaggelos, A. K. Digital image restoration. *IEEE signal processing magazine*, 14(2): 24–41, 1997.

Bansal, A., Borgnia, E., Chu, H.-M., Li, J. S., Kazemi, H., Huang, F., Goldblum, M., Geiping, J., and Goldstein, T. Cold diffusion: Inverting arbitrary image transforms without noise. *arXiv preprint arXiv:2208.09392*, 2022.

Baraniuk, R. G. Compressive sensing [lecture notes]. *IEEE signal processing magazine*, 24(4):118–121, 2007.

Besag, J., York, J., and Mollié, A. Bayesian image restoration, with two applications in spatial statistics. *Annals of the institute of statistical mathematics*, 43(1):1–20, 1991.

Bigdeli, S., Honzátko, D., Süsstrunk, S., and Dunbar, L. A. Image restoration using plug-and-play cnn map denoisers. *arXiv preprint arXiv:1912.09299*, 2019.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Brock, A., Donahue, J., and Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Buades, A., Coll, B., and Morel, J.-M. A non-local algorithm for image denoising. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, pp. 60–65. Ieee, 2005.

Castleman, K. R. *Digital image processing*. Prentice Hall Press, 1996.

Chen, T., Liu, G.-H., and Theodorou, E. A. Likelihood training of schrödinger bridge using forward-backward sdes theory. *The International Conference on Learning Representations*, 2022.

Chen, Y. Higher-order mrfs based image super resolution: why not map? *IET Image Processing*, 10(4):297–303, 2016.

Daras, G., Delbracio, M., Talebi, H., Dimakis, A. G., and Milanfar, P. Soft diffusion: Score matching for general corruptions. *arXiv preprint arXiv:2209.05442*, 2022.

De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.

Denton, E. L., Chintala, S., Fergus, R., et al. Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in neural information processing systems*, 28, 2015.

Dockhorn, T., Vahdat, A., and Kreis, K. Score-based generative modeling with critically-damped langevin diffusion. *The International Conference on Learning Representations*, 2022a.

Dockhorn, T., Vahdat, A., and Kreis, K. Genie: Higher-order denoising diffusion solvers. *Advances in Neural Information Processing Systems*, 2022b.

Donoho, D. L. De-noising by soft-thresholding. *IEEE transactions on information theory*, 41(3):613–627, 1995.

Dziugaite, G. K., Roy, D. M., and Ghahramani, Z. Training generative neural networks via maximum mean discrepancy optimization. *arXiv preprint arXiv:1505.03906*, 2015.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Farsiu, S., Robinson, M. D., Elad, M., and Milanfar, P. Fast and robust multiframe super resolution. *IEEE transactions on image processing*, 13(10):1327–1344, 2004.

Gao, R., Song, Y., Poole, B., Wu, Y. N., and Kingma, D. P. Learning energy-based models by diffusion recovery likelihood. *Advances in neural information processing systems*, 2021.

Geman, D. and Yang, C. Nonlinear image recovery with half-quadratic regularization. *IEEE transactions on Image Processing*, 4(7):932–946, 1995.

Gong, X., Chang, S., Jiang, Y., and Wang, Z. Autogan: Neural architecture search for generative adversarial networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3224–3234, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Grathwohl, W., Chen, R. T., Bettencourt, J., Sutskever, I., and Duvenaud, D. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*, 2018.

Gunturk, B. and Li, X. *Image restoration*. CRC Press, 2018.

Hadamard, J. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton university bulletin*, pp. 49–52, 1902.

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

Hoogeboom, E. and Salimans, T. Blurring diffusion models. *arXiv preprint arXiv:2209.05557*, 2022.

Hunt, B. R. Bayesian methods in nonlinear digital image restoration. *IEEE Transactions on Computers*, 26(03): 219–229, 1977.

Hurault, S., Leclaire, A., and Papadakis, N. Gradient step denoiser for convergent plug-and-play. *arXiv preprint arXiv:2110.03220*, 2021.

Hurault, S., Leclaire, A., and Papadakis, N. Proximal denoiser for convergent plug-and-play optimization with nonconvex regularization. *arXiv preprint arXiv:2201.13256*, 2022.

Jiang, Y., Chang, S., and Wang, Z. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*, 1(3), 2021.

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

Jolicoeur-Martineau, A., Li, K., Piché-Taillefer, R., Kachman, T., and Mitliagkas, I. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021a.

Jolicoeur-Martineau, A., Li, K., Piché-Taillefer, R., Kachman, T., and Mitliagkas, I. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021b.

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

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

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020.

Kawar, B., Vaksman, G., and Elad, M. Snips: Solving noisy inverse problems stochastically. *Advances in Neural Information Processing Systems*, 34:21757–21769, 2021.

Kawar, B., Elad, M., Ermon, S., and Song, J. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 2022.

Kim, D., Shin, S., Song, K., Kang, W., and Moon, I.-C. Score matching model for unbounded data score. *arXiv preprint arXiv:2106.05527*, 2021.

Kim, D., Shin, S., Song, K., Kang, W., and Moon, I.-C. Soft truncation: A universal training technique of score-based diffusion model for high precision score estimation. In *International Conference on Machine Learning*, pp. 11201–11228. PMLR, 2022.

Kingma, D., Salimans, T., Poole, B., and Ho, J. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *International conference on machine learning*, 2014.

Kong, Z. and Ping, W. On fast sampling of diffusion probabilistic models. *arXiv preprint arXiv:2106.00132*, 2021.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Laumont, R., Bortoli, V. D., Almansa, A., Delon, J., Durmus, A., and Pereyra, M. Bayesian imaging using plug & play priors: when langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022.

Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018.

Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y., and Póczos, B. Mmd gan: Towards deeper understanding of moment matching network. *Advances in neural information processing systems*, 30, 2017.

Li, Y., Swersky, K., and Zemel, R. Generative moment matching networks. In *International conference on machine learning*, pp. 1718–1727. PMLR, 2015.

Lions, P.-L. and Mercier, B. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

Liu, L., Ren, Y., Lin, Z., and Zhao, Z. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 2022.

Lunz, S., Öktem, O., and Schönlieb, C.-B. Adversarial regularizers in inverse problems. *Advances in neural information processing systems*, 31, 2018.

Lyu, Z., Xu, X., Yang, C., Lin, D., and Dai, B. Accelerating diffusion models via early stop of the diffusion process. *arXiv preprint arXiv:2205.12524*, 2022.

Ma, J., Huang, J., Feng, Q., Zhang, H., Lu, H., Liang, Z., and Chen, W. Low-dose computed tomography image restoration using previous normal-dose scan. *Medical physics*, 38(10):5713–5731, 2011.

Mallat, S. *A wavelet tour of signal processing*. Elsevier, 1999.

Meng, C., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Moore, E. H. On the reciprocal of the general algebraic matrix. *Bull. Am. Math. Soc.*, 26:394–395, 1920.

Neal, R. M. *Probabilistic inference using Markov chain Monte Carlo methods*. Department of Computer Science, University of Toronto Toronto, ON, Canada, 1993.

Nguyen, K., Ho, N., Pham, T., and Bui, H. Distributional sliced-wasserstein and applications to generative modeling. *arXiv preprint arXiv:2002.07367*, 2020.

Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C. C., and Luo, P. Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Pandey, K., Mukherjee, A., Rai, P., and Kumar, A. Diffusevae: Efficient, controllable and high-fidelity generation from low-dimensional latents. *arXiv preprint arXiv:2201.00308*, 2022.

Parikh, N., Boyd, S., et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014.

Parmar, G., Li, D., Lee, K., and Tu, Z. Dual contradistinctive generative autoencoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 823–832, 2021.

Pidhorskyi, S., Adjeroh, D. A., and Doretto, G. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14104–14113, 2020.

Reed, S., Oord, A., Kalchbrenner, N., Colmenarejo, S. G., Wang, Z., Chen, Y., Belov, D., and Freitas, N. Parallel multiscale autoregressive density estimation. In *International conference on machine learning*, pp. 2912–2921. PMLR, 2017.

Reehorst, E. T. and Schniter, P. Regularization by denoising: Clarifications and new interpretations. *IEEE transactions on computational imaging*, 5(1):52–67, 2018.

Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Rissanen, S., Heinonen, M., and Solin, A. Generative modelling with inverse heat dissipation. *arXiv preprint arXiv:2206.13397*, 2022.

Romano, Y., Elad, M., and Milanfar, P. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Rudin, L. I., Osher, S., and Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

Saha, S., Boers, Y., Driessen, H., Mandal, P. K., and Bagchi, A. Particle based map state estimation: A comparison. In *2009 12th International Conference on Information Fusion*, pp. 278–283. IEEE, 2009.

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

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *Advances in Neural Information Processing Systems*, 2021a.

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

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *The International Conference on Learning Representations*, 2021b.

Tachibana, H., Go, M., Inahara, M., Katayama, Y., and Watanabe, Y. It\^{o}-taylor sampling scheme for denoising diffusion probabilistic models using ideal derivatives. *arXiv preprint arXiv:2112.13339*, 2021.

Tikhonov, A. N. On the regularization of ill-posed problems. In *Doklady Akademii Nauk*, volume 153, pp. 49–52. Russian Academy of Sciences, 1963.

Trussell, H. The relationship between image restoration by the maximum a posteriori method and a maximum entropy method. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(1):114–117, 1980.

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

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

Van Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. Pixel recurrent neural networks. In *International conference on machine learning*, pp. 1747–1756. PMLR, 2016.

Venkatakrishnan, S. V., Bouman, C. A., and Wohlberg, B. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pp. 945–948. IEEE, 2013.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Xiao, Z., Kreis, K., Kautz, J., and Vahdat, A. Vaebm: A symbiosis between variational autoencoders and energy-based models. *arXiv preprint arXiv:2010.00654*, 2020.

Xiao, Z., Kreis, K., and Vahdat, A. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021a.

Xiao, Z., Yan, Q., and Amit, Y. Ebms trained with maximum likelihood are generator models trained with a self-adverserial loss. *arXiv preprint arXiv:2102.11757*, 2021b.

Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Zervakis, M. and Venetsanopoulos, A. A class of noniterative estimators for nonlinear image restoration. *IEEE transactions on circuits and systems*, 38(7):731–744, 1991.

Zhang, B., Gu, S., Zhang, B., Bao, J., Chen, D., Wen, F., Wang, Y., and Guo, B. Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11304–11314, 2022.

Zhang, Q. and Chen, Y. Diffusion normalizing flow. *Advances in Neural Information Processing Systems*, 34: 16280–16291, 2021.

Zhang, Q. and Chen, Y. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022.

A. Continuation of Related Works

DDMs have been pertinent generative models by showing promising results on various generation tasks. DDMs degrade the data with a reference diffusion process and learn the data distribution by restoring it. We have arranged DDMs in the context of restoration, and DDMs can be interpreted as an MMSE estimator for a denoising task.

- Energy-based models (EBMs) are another line of generative models that learn the unnormalized data distribution by giving low energy to high-density regions in the data space. As DDMs have demonstrated that recovery of a sequence of noisy data is more effective than directly approximating the data density, Gao et al. (2021) recently proposed a recovery energy-based model (REBM) by using a diffusion process. Inspired by DDMs, REBM learns a sequence of energy functions for the marginal distributions of the diffusion process. More precisely, from the noisy observation $\tilde{\mathbf{x}} = \mathbf{x} + \xi,\ \xi \sim \mathcal{N}\left(0, \sigma^2 I\right)$, they estimate the conditional likelihood $p_\theta\left(\mathbf{x} \mid \tilde{\mathbf{x}}\right) \propto \exp^{-\mathcal{E}_\theta(\mathbf{x}|\tilde{\mathbf{x}})}$ by learning the energy function
$$\mathcal{E}_\theta = \frac{1}{2\sigma^2}\left\|\mathbf{x} - \tilde{\mathbf{x}}\right\|^2 - f_\theta\left(\mathbf{x}\right). \tag{9}$$
They indeed learn the marginal density f_θ and infer the data through the recovery likelihood. The marginal density f_θ is adversarially trained by assigning low energy to high-probability regions in the data space and high energy values outside these regions. Since direct sampling from $p_\theta\left(\mathbf{x} \mid \tilde{\mathbf{x}}\right)$ is intractable, samples are usually drawn by leveraging Langevin dynamics (LD) (Neal, 1993), which is a conventional sampling method of EBMs. Therefore, REBM trains marginal density f_θ using a kind of adversarial loss, but REBM is actually a MAP estimator implicitly defined by the sampling dynamics. In other words, REBM learns the posterior distribution using the reference diffusion process, but it does not deviate from the traditional sampling method of EBM, still generating samples through inefficient LD. There are two difficulties of such a Markov Chain Monte Carlo (MCMC) sampling: Applying MCMC in pixel space to sample one instance from the model is impractical due to the high dimensionality and long inference time. As reported in (Xiao et al., 2021b), the estimated density of EBMs can sometimes differ significantly from the data distribution, even if the model with the short-run LD produces relevant samples. It is also known that the convergence of LD is very difficult when the energy function is complicated.

- Another related work is a denoising diffusion GAN (DDGAN) (Xiao et al., 2021a), which enjoys small sampling steps by using GAN. DDGAN focuses on improving the sampling efficiency while maintaining the sample quality and mode coverage of DDMs. The reason why DDMs adhere to the heavy sampling scheme is their common assumption that the true posterior is approximated by Gaussian distributions. This assumption holds only with small denoising steps. When the number of denoising steps is reduced, the denoising distribution is no longer a Gaussian distribution, but a non-Gaussian multi-modal, which is usually intractable. DDGAN breaks the Gaussian assumption by reducing the number of denoising steps and then approximates the non-Gaussian multimodal posterior distribution with the help of GAN. DDGAN enhances the sampling efficiency of DDMs and also resolves the mode collapse problem of GANs by using a couple of denoising steps from the perspective of GAN literature. The architecture of DDGAN is somewhat similar to that of our RGM-KLD-D. However, there is a difference in a way of estimating MAP. DDGAN assigns all responsibility for MAP estimation to the GAN structure. On the other hand, our models learn the MAP-based estimator by separating the posterior distribution into the fidelity term and the prior term. Therefore, the model is much easier to learn than DDGAN. As a consequence, RGM-KLD-D obtains substantial savings in terms of training iterations than DDGAN. Specifically, in CIFAR10 experiments, DDGAN takes 400K iterations to achieve FID of 3.75. In comparison, our RGM-KLD-D only uses 150K iterations to achieve the same performance as DDGAN and takes 200K iterations for FID of 3.04. For the CelebA dataset, DDGAN requires 750K iterations to attain FID 7.64, while RGM-KLD-D obtains the same FID score using only 450K iterations and FID 7.15 even with 500K iterations. Furthermore, our framework can be extended to various forward processes and regularization terms, which are more flexible and utilizable.

As such, there have been various density estimation models based on denoising. Diffusion models, such as DDPM and score matching with Langevin dynamics and its variants, are MMSE-based estimators. The model of REBM itself approximates the marginal density as we do, but our model is trained with MAP-based loss, whereas REBM generates samples from the posterior distribution through the sampling method. Diffusion models and REBM train different estimators, but both models use a Langevin sampling scheme that requires thousands of network evaluations. On the other hand, DDGAN is a model that can perform one-shot sampling with the help of GAN (away from the Langevin sampling), just like our RGMs. However, since DDGAN learns the whole posterior density through the discriminator, it is more inefficient in terms of learning than our models, which separate the fidelity and the prior term. Consequently, our RGMs achieve better performance than DDGAN with much fewer iterations. All these models are restricted to the diffusion process. Otherwise, our RGMs can

enjoy flexible forward processes and are also given a degree of freedom in how to parametrize the prior term. In other words, our approach does not need to restrict to the diffusion process and unlike DDGAN, which is limited to the GAN structure, it is possible to design the prior term by leveraging different generation models. This is further discussed in Appendix B.2.

B. Implementation Details

B.1. Degradation Schedule

Let $\mathbf{A}_k$ and $\mathbf{\Sigma}_k$ be a degradation matrix and a noise variance on the k-th degradation step, respectively.

Then, given a data $\mathbf{x}$ sampled from the real data distribution p_{data}, a degraded data $\mathbf{y}_k$ on the k-th forward step is sampled from

$$p\left(\mathbf{y}_k \mid \mathbf{x}\right) = \mathcal{N}\left(\mathbf{y}_k; \mathbf{A}_k\mathbf{x}, \mathbf{\Sigma}_k\right).$$

We denote the marginal distribution at the T-th degradation step as p_T. Because our primary goal is to bridge p_{data} to an easy-to-sample distribution p_T, (especially to a zero mean Gaussian distribution), we gradually decrease the norm of $\mathbf{A}_k$ to zero as k increases. In Section 4, we introduced two families of models based on the degradation schedule $\{(\mathbf{A}_k, \mathbf{\Sigma}_k)\}_{k=1}^T$ with the corner cases: RGM-KLD-D for $\mathbf{A}_k = \mathbf{I}$ and RGM-SR for $\mathbf{A}_k = \mathbf{P}_k$ a 2×2 averaging filter. Roughly speaking, we consider three models based on different forward processes designed as follows:

- RGM-D: noise $\rightarrow$ noise $\rightarrow$ noise $\rightarrow$ noise $\rightarrow \cdots$.
- RGM-SR (naive): downsample + noise $\rightarrow$ downsample + noise $\rightarrow \cdots$.
- RGM-SR: noise $\rightarrow$ downsample $\rightarrow$ noise $\rightarrow$ downsample $\rightarrow \cdots$,

shown schematically in Figure 8. With the following notations

$$\beta_k = \frac{1}{4}\left(\beta_{\max} - \beta_{\min}\right)\left(\frac{k}{T}\right)^2 + \frac{1}{2}\beta_{\min}\frac{k}{T}, \tag{10}$$

$$\tilde{\beta}_k = \frac{1}{4}\left(\beta_{\max} - \beta_{\min}\right)\left(\frac{k}{T}\right)^4 + \frac{1}{2}\beta_{\min}\left(\frac{k}{T}\right)^2, \tag{11}$$

where $\beta_{\max} = 20$ and $\beta_{\min} = 0.1$, table 5 details the explicit form of the forward processes used for each model. The noise

Table 5: The choice of schedule $\mathbf{A}_k$ and $\mathbf{\Sigma}_k$ and the corresponding latent distribution p_T for RGM-D, RGM-SR (naive), and RGM-SR. $\mathbf{P}_k$ is a projection matrix that downscales the images by block averaging in a factor of 2^k. For RGM-SR, we set T in (10) to be half of the total steps added by one.

	RGM-D	RGM-SR (naive)	RGM-SR
$\mathbf{A}_k$	$e^{-\beta_k}\mathbf{I}$	$e^{-\tilde{\beta}_k}\mathbf{P}_k$	$e^{-\beta_{\lceil k/2 \rceil}}\mathbf{P}_{\lfloor k/2 \rfloor}$
$\mathbf{\Sigma}_k$	$\left(1 - e^{-2\beta_k}\right)^2\mathbf{I}$	$\left(1 - e^{-2\tilde{\beta}_k}\right)^2\mathbf{P}_k^\top\mathbf{P}_k$	$\left(2^{\lceil k/2 \rceil}\left(1 - e^{-2\beta_{\lceil k/2 \rceil}}\right)\right)^2\mathbf{P}_{\lfloor k/2 \rfloor}^\top\mathbf{P}_{\lfloor k/2 \rfloor}$
p_T	$\mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$	$\mathcal{N}\left(\mathbf{0}, \frac{1}{64}\mathbf{I}\right)$	$\mathcal{N}\left(\mathbf{0}, 4\mathbf{I}\right)$

schedule of RGM-D follows the Variance Preserving SDE provided bySong et al. (2021b), and others are implemented with a slight modification of them.

When we use the degradation matrix $\mathbf{A}_k$ as the averaging filter, the corresponding forward process downsamples the image while adding Gaussian noise. RGM according to this forward process, referred to as RGM-SR (naive), is demanded to super-resolve the degraded data while simultaneously denoising it. It is considerably more difficult than the denoising task when the noise level is the same. To address this difficulty, we consider a newly scheduled degradation scheme that decomposes the forward process into downsampling and noising operations. We name the RGM designed in conjunction with this forward schedule as RGM-SR. As provided in Table 5, when the step k is odd, the difference from the $(k+1)$-th step is only the projection matrix. Namely, only downsample is performed when sampling the $(k+1)$-th degraded data from the k-th degraded observation. Conversely, when k is an even number, the forward process produces the $(k+1)$-th degraded image by adding the Gaussian noise. In summary, RGM-SR focuses on denoising the data in odd steps and super-resolving

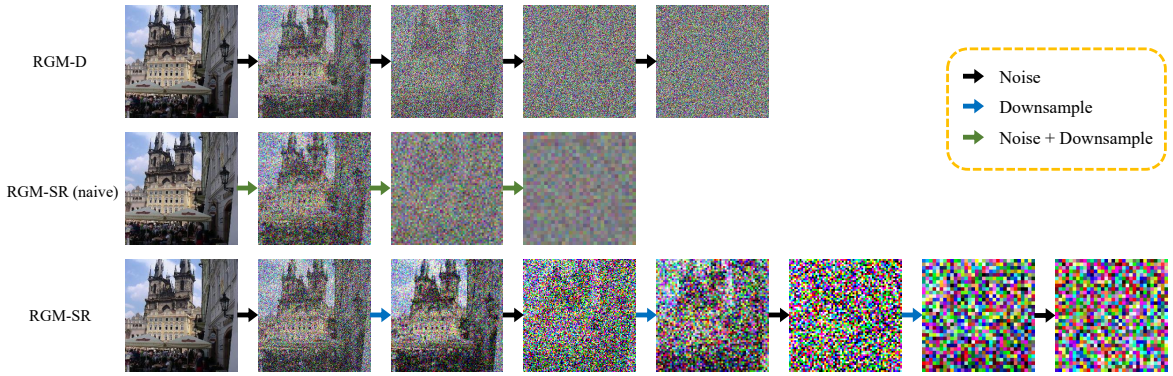

Figure 8: Degradation sequences for RGMs.

the data in even steps. Provably due to the difficulty of performing super-resolution and denoising simultaneously, RGM-SR (naive) has the worst performance. Whereas RGM-SR, which uses the decomposed forward process, outperforms both RGM-D and RGM-SR by a large margin as reported in Section 4.2.

B.2. Training RGMs

In this section, we unambiguously elucidate how we train our RGMs. In Algorithms 1 and 2, we summarize the two training procedures of GAN-based prior that are suited to different situations. We also provide an explanation of the training procedure of RGMs with other priors. Moreover, the generation process is provided in Algorithm 3.

Training As proposed in Section 3.1, RGMs learn the data distribution p_{data} through the process of degrading the image through a forward process and then restoring it using the MAP-based objective (6). However, since it is too difficult to restore the image directly from the Gaussian distribution in one shot, we use a handful of forward steps and train RGMs that recover the distribution between each step. (We also include an ablation study on this in Appendix C.1) In other words, at each step k, we first sample a degraded image $\mathbf{y}_k$ of a given image $\mathbf{x} \sim p_{\text{data}}$. The generator G_θ generates the restored image $\hat{\mathbf{x}}$, and then, we degrade it by the posterior distribution $\hat{\mathbf{y}}_{k-1} \sim p\left(\hat{\mathbf{y}}_{k-1} \mid \mathbf{y}_k, \hat{\mathbf{x}}\right)$. We train our loss function so that $\hat{\mathbf{y}}_{k-1}$ becomes a restoration of $\mathbf{y}_k$. The discriminator loss is also imposed on the $(k-1)$-th step. Through the overall process, we ultimately learn the model that restores the distribution of the previous $(k-1)$-th step at each k-th step.

The training procedure is articulated in Algorithm 1.

However, we can exactly formulate the posterior distribution only when the forward process satisfies certain conditions. For all $k = 1, \cdots, T$, if there exists $\left(\tilde{\mathbf{A}}_k, \tilde{\mathbf{\Sigma}}_k\right)$ satisfying

$$\mathbf{A}_k = \tilde{\mathbf{A}}_k \mathbf{A}_{k-1}, \ \tilde{\mathbf{\Sigma}}_k := \mathbf{\Sigma}_k - \tilde{\mathbf{A}}_k \mathbf{\Sigma}_k \tilde{\mathbf{A}}_k^\top \succ 0, \tag{12}$$

we can explicitly construct a conditional distribution $p_{k|k-1}(\mathbf{y}_k|\mathbf{y}_{k-1}) = \mathcal{N}(\tilde{\mathbf{A}}_k \mathbf{y}_{k-1}, \tilde{\mathbf{\Sigma}}_k)$ and a posterior distribution (Ho et al., 2020; Kingma et al., 2021; Xiao et al., 2021a). For example, the forward process of RGM-D falls under this condition (12), but that of RGM-SR does not. Therefore, the Algorithm 1 does not fit with RGM-SR. To unravel such a problem, we propose a prevalent algorithm that is applicable to forward processes that are in discord with the condition (12). See Algorithm 2. The only difference from the Algorithm 1 is the replacement of the posterior sampling by the prior sampling in Line 5 and the data fidelity term in Line 9. When the posterior distribution is unavailable, we corrupt the image $\hat{\mathbf{x}}$ restored by the generator G_θ to the $(k-1)$-th degraded distribution using the k-th forward process rather than posterior sampling. Moreover, since the conditional distribution between k and $(k-1)$ steps is unknown, we adopt the fidelity term of the image $\hat{\mathbf{x}}$ reconstructed by the generator. This algorithm is universally applicable to general forward processes. One notable fact is that RGM-D, whose posterior distribution is tractable, learns the data distribution better when using this algorithm than Algorithm 1. This is discussed in detail in Appendix C.1.

Training with other priors Without being tied to the GAN discriminator, our RGMs have the freedom to parametrize the prior term g of regularizer (6) in any other way. To demonstrate that the RGM framework universally works for variously

Algorithm 1 Training of RGMs with Posterior sampling

Input: Dataset $\mathcal{D}$, degradation schedule $\{(\mathbf{A}_k, \mathbf{\Sigma}_k)\}_{k=0}^{T}$ with $(\mathbf{A}_0, \mathbf{\Sigma}_0) = (\mathbf{I}, \mathbf{0})$, posterior distribution $p_{k|k-1}(\mathbf{y}_{k-1}, \mathbf{y}_k) = \mathcal{N}\left(\tilde{\mathbf{A}}_k \mathbf{y}_k, \tilde{\mathbf{\Sigma}}_k\right)$, generator G_θ, discriminator D_ϕ, and regularization parameter $\lambda \geq 0$.

1: **for** $i = 0, 1, 2, \ldots$ **do**
2: Sample data $\mathbf{x} \in \mathcal{D}$.
3: Sample $k \sim \mathrm{Uniform}(\{1, 2, \ldots, T\})$.
4: Sample $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$.
5: Sample degraded data $\mathbf{y}_k \sim \mathcal{N}(\mathbf{A}_k \mathbf{x}, \mathbf{\Sigma}_k)$ and $\mathbf{y}_{k-1} \sim \mathcal{N}(\mathbf{A}_{k-1} \mathbf{x}, \mathbf{\Sigma}_{k-1})$.
6: Generate an image $\hat{\mathbf{x}} = G_\theta(\mathbf{y}_k, k, \mathbf{z})$.
7: Degrade data by posterior sampling $\hat{\mathbf{y}}_{k-1} \sim p(\hat{\mathbf{y}}_{k-1} \mid \mathbf{y}_k, \hat{\mathbf{x}})$.
8: Update ϕ by the following loss:

$$\log\left(1 - D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right)\right) + \log D_\phi\left(\mathbf{y}_{k-1}, k-1\right).$$

9: Update θ by the following loss:

$$\log\left(1 - D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right)\right) - \log D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right) + \frac{1}{2\lambda}\left\|\left(\tilde{\mathbf{\Sigma}}_k^{\dagger}\right)^{\frac{1}{2}}\left(\tilde{\mathbf{A}}_k \hat{\mathbf{y}}_{k-1} - \mathbf{y}_k\right)\right\|_2^2.$$

10: **end for**

Algorithm 2 Relaxed training algorithm of RGMs

Input: Dataset $\mathcal{D}$, degradation schedule $\{(\mathbf{A}_k, \mathbf{\Sigma}_k)\}_{k=0}^{T}$ with $(\mathbf{A}_0, \mathbf{\Sigma}_0) = (\mathbf{I}, \mathbf{0})$, discriminator D_ϕ, generator G_θ, and regularization parameter $\lambda \geq 0$.

1: **for** $i = 0, 1, 2, \ldots$ **do**
2: Sample $\mathbf{x} \in \mathcal{D}$.
3: Sample $k \sim \mathrm{Uniform}(\{1, 2, \ldots, T\})$.
4: Sample $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$.
5: Sample degraded data $\mathbf{y}_k \sim \mathcal{N}(\mathbf{A}_k \mathbf{x}, \mathbf{\Sigma}_k)$ and $\mathbf{y}_{k-1} \sim \mathcal{N}(\mathbf{A}_{k-1} \mathbf{x}, \mathbf{\Sigma}_{k-1})$.
6: Generate an image $\hat{\mathbf{x}} = G_\theta(\mathbf{y}_k, k, \mathbf{z})$.
7: Degrade $\hat{\mathbf{x}}$ by $\hat{\mathbf{y}}_{k-1} \sim \mathcal{N}(\mathbf{A}_{k-1} \hat{\mathbf{x}}, \mathbf{\Sigma}_{k-1})$.
8: Update ϕ by the following loss:

$$\log\left(1 - D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right)\right) + \log D_\phi\left(\mathbf{y}_{k-1}, k-1\right).$$

9: Update θ by the following loss:

$$\log\left(1 - D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right)\right) - \log D_\phi\left(\hat{\mathbf{y}}_{k-1}, k-1\right) + \frac{1}{2\lambda}\left\|\left(\mathbf{\Sigma}_k^{\dagger}\right)^{\frac{1}{2}}\left(A_k \hat{\mathbf{x}} - \mathbf{y}_k\right)\right\|_2^2.$$

10: **end for**

parametrized prior terms, we design the prior term in two additional ways: *maximum mean discrepancy (MMD)* (Dziugaite et al., 2015) and *distributed sliced Wasserstein distance (DSWD)* (Nguyen et al., 2020):

- We replace KLD objective to MMD, a two-sample test based on kernel maximum mean discrepancy (Li et al., 2017). For given two sets of data $X = \{x_1, x_2 \ldots, x_M\}$ and $Y = \{y_1, y_2 \ldots, y_M\}$, the MMD prior $g(X, Y)$, which estimates the MMD distance, is defined as follows;

$$g\left(X, Y\right) = \frac{1}{\binom{M}{2}}\left[\sum_{i \neq j} k\left(x_i, x_j\right) - 2\sum_{i \neq j} k\left(x_i, y_j\right) + \sum_{i \neq j} k\left(y_i, y_j\right)\right], \tag{13}$$

  where k is a positive definite kernel. Following the prior works (Dziugaite et al., 2015; Li et al., 2015; 2017), we use a mixture of RBF kernels $k(x, x') = \sum_{i=1}^{n} k_{\sigma_i}(x, x')$ where k_σ is a Gaussian kernel with bandwidth parameter of σ.

- To measure the distance of two datasets $X = \{x_1, x_2 \ldots, x_M\}$ and $Y = \{y_1, y_2 \ldots, y_M\}$, sliced Wasserstein-based framework (SW) projects the data into a one-dimensional vector then explicitly calculates the Wasserstein distance on the projected space. In such an explicit calculation, SW can be freed from an unstable adversarial framework. Recently, Nguyen et al. (2020) has proposed a novel and efficient method to obtain useful projection samples, hence, we followed the implementation of this prior work in our experiments. Specifically, following Nguyen et al. (2020), we use the learnable feature function and calculate DSWD on the feature space for CIFAR10 experiments. In other words, we replace the prior term g of line 9 of Algorithm 2 to DSWD objective.

The results on the 2D synthetic example discussed in Figure 3 validate that RGMs parametrized in three different ways show consistent performance, where they are all more efficient than the MMSE estimator. Furthermore, we also carried out the experiment of RGM-D with the DSWD prior, termed RGM-DSWD-D, on CIFAR10. Consequently, RGM-DSWD-D achieves an FID score of 3.14 retaining comparable performance with RGM-KLD-D. The overall results verify that our MAP approach works universally well for the various prior terms.

Sampling The sampling algorithm is summarized in Algorithm 3. Starting from a latent variable $\mathbf{y}_T \sim p_T$, the trained G_θ generates the restored image $\tilde{\mathbf{x}} = G_\theta\left(\mathbf{y}_{k+1}, k, \mathbf{z}\right)$ with a randomly selected auxiliary variable $\mathbf{z}$ from the $(k+1)$-the degraded image $\mathbf{y}_{k+1}$, and then corrupt it by passing the k-th forward process. Continue this procedure until $k = 0$. When we train our model with Algorithm 1, line 5 should be replaced by the posterior sampling. For a schematic representation of this hierarchical sampling process of RGMs, see Figure 9.

Algorithm 3 Sampling Procedure of RGMs

Input: Trained generator G_θ and degradation schedule $\{\mathbf{A}_k, \mathbf{\Sigma}_k\}_{k=1}^{T}$.
1: Sample initial state $\mathbf{y}_T \sim \mathcal{N}\left(0, \mathbf{\Sigma}_T\right)$.
2: **for** $k = T-1, T-2, \ldots, 0$ **do**
3: Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
4: Restore image $\hat{\mathbf{x}}_k$ by $\hat{\mathbf{x}}_k = G_\theta\left(\mathbf{y}_{k+1}, k+1, \mathbf{z}\right)$.
5: Sample $\mathbf{y}_k \sim \mathcal{N}\left(\mathbf{A}_k \hat{\mathbf{x}}_k, \mathbf{\Sigma}_k\right)$.
6: **end for**
7: **return** $\hat{\mathbf{x}}_0$

B.3. Implementation Details

We refer to (Nguyen et al., 2020) for the precise definition of hyperparameters of RGM-DSWD-D.

Experiments on 2D dataset In the implementation of the two-dimensional Gaussian Mixture, we use a 3-layered MLP of 32 hidden dimensions for both generator and discriminator with Tanh activation. We concatenated all the inputs and passed them through the network. For the RGM-KLD-D experiment, models are trained for 100K iterations with a learning rate of 10^{-4}, a batch size of 1000. For the RGM-DSWD-D experiment on the 2D data, we use the number of iterations of 100K, the number of projections of 10, 10 DSW iterations, and $\lambda_C = 10$. For the MMD experiment, we applied kernel bandwidths of 0.1, 0.5, 1, 2, and 10.

Image generation To optimize our RGMs, we mostly followed the previous literature (Xiao et al., 2021a), including network architectures, R_1 regularization, and optimizer settings. Note that our code is largely built on top of DDGAN [1]

[1] https://github.com/NVlabs/denoising-diffusion-gan

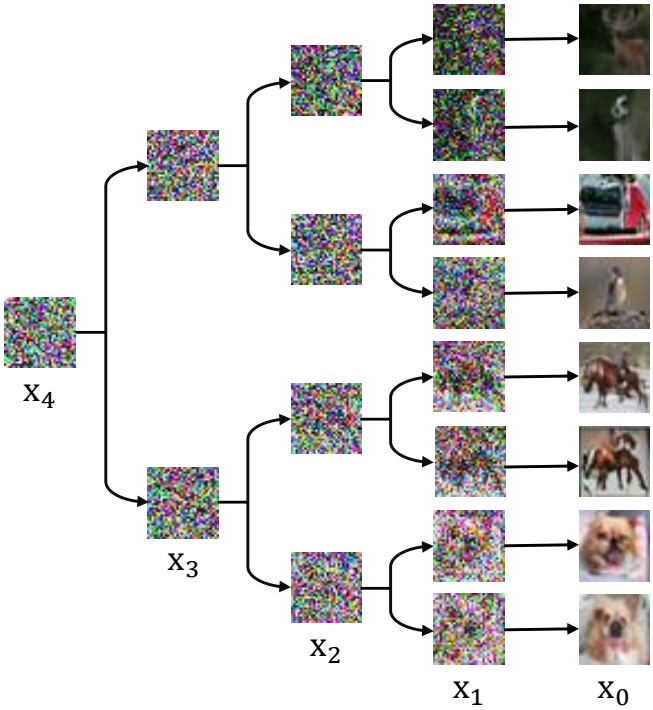

Figure 9: Hierarchical generation process of RGMs.

(MIT License). We vary the discriminator by simply changing input channels into three. Moreover, we use a learning rate of 2×10^{-4} for generator update in all experiments and a learning rate of 10^{-4} for discriminator update. We use $\lambda^{-1} = 10^{-3}$ for image size of 32, and $\lambda^{-1} = 5 \times 10^{-5}$ for image size of 256. The models are trained with Adam (Kingma & Ba, 2014) in all experiments. In CIFAR10 experiments, we train RGM-KLD-D and RGM-KLD-SR (naive) for 200K iterations and RGM-KLD-SR for 230K iterations. Moreover, for RGM-DSWD-D implementation on CIFAR10, we use the output of the fifth convolutional layer of the discriminator as a feature vector. We use the number of iterations of 150K, the number of projections of 1000, 10 DSW iterations, and $\lambda_C = 1$ for the DSWD experiment. Lastly, we train RGM-KLD-D for 500K iterations and 300K iterations in LSUN experiments.

Other details We train our models on CIFAR-10 using 4 V100 GPUs. The training takes approximately 40 hours on CIFAR-10. Moreover, the sampling of 100 samples takes approximately 0.25 seconds for RGM-KLD-D on single V100 GPUs. For evaluation on CIFAR10, we use 50K generated samples to measure IS and FID. For CelebA-HQ-256, we use 30K samples to compute FID.

B.4. Solving Inverse Problems

Modern image processing algorithms reconstruct the ground-truth image by solving the following minimization problem:

$$\underset{\mathbf{x}}{\text{minimize}}\ f_{\mathbf{y}}(\mathbf{x}) + \lambda g(\mathbf{x}),$$

where f measures the fidelity to a corrupted observation $\mathbf{y}$, and g constrains the solution space by measuring the complexity or noisiness of the image. Many imaging inverse problems, such as colorization, super-resolution (SR), and deblurring, fall under this form. Since the above optimization problem does not have a closed-form solution in general, first-order proximal splitting algorithms, including half-quadratic splitting (HQS) (Geman & Yang, 1995), alternating direction method of multipliers (ADMM) (Boyd et al., 2011), solve the problem by operating individually on f and g via the proximal operator (Parikh et al., 2014). With the aid of the emergence of deep learning, Plug-and-Play (PnP) algorithms (Venkatakrishnan et al., 2013) have recently begun to connect proximal splitting algorithms and deep neural networks by replacing the proximity operator of the regularization term g with a generic denoiser (Romano et al., 2017; Reehorst & Schniter, 2018).

Similarly, our trained RGMs can be used as PnP priors. In Section 4.4 we solved two inverse problems, colorization, and super-resolution, by plugging the trained RGMs into Douglas-Rachford Splitting (DRS) algorithm (Lions & Mercier, 1979),

following (Hurault et al., 2022). This is summarized in Algorithm 4. Starting from the degraded observation $\mathbf{y}$, the DRS algorithm updates the solution by alternatively utilizing proximal operations for both f and g. By iteratively updating the solution, the solution lies far outside the distribution on which our denoiser G_θ trained. For this out-distribution data, G_θ cannot recover the original image distribution, which in turn prevents the DRS algorithm from convergence. To remedy this problem, the input of G_θ should always be within the trained distribution. Therefore, we push the updated solution into the learned distribution through the forward process. Note that the proximal operation is calculated by utilizing efficient singular value decomposition proposed in Kawar et al. (2022).

Algorithm 4 Solving Inverse Problems by RGMs

Input: A degraded observation $\mathbf{y}$, fidelity loss function $f_{\mathbf{y}}$, repeat number M, update rate $\alpha \in (0, 1]$, regularization parameter $\lambda \geq 0$, trained generator G_θ, and degradation schedule $\{\mathbf{A}_k, \mathbf{\Sigma}_k\}_{i=1}^T$.

Initialize $\mathbf{x}_K = \mathbf{y}$.
for $0, 1, \ldots, M$ **do**
  for $i = K, K-1, \ldots, 1$ **do**
    Sample $\hat{\mathbf{y}} \sim \mathcal{N}(\mathbf{A}\mathbf{x}_i, \mathbf{\Sigma}_i)$ and $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$.
    $\hat{\mathbf{x}} \leftarrow G_\theta(\hat{\mathbf{y}}, i-1, \mathbf{z})$.
    $\hat{\mathbf{x}} \leftarrow (1-\alpha)\mathbf{x}_i + \alpha\hat{\mathbf{x}}$.
    $\Delta\mathbf{x} \leftarrow \mathrm{prox}_{\lambda f_{\mathbf{y}}}(2\hat{\mathbf{x}} - \mathbf{x}_i) - \hat{\mathbf{x}}$.
    $\mathbf{x}_{i-1} \leftarrow \mathbf{x}_i + \Delta\mathbf{x}$.
  end for
end for
return $\mathbf{x}_0$

Settings & Hyperparameters In SR experiments, we downscale images by using a block averaging filter by r in each axis. The filter is applied for the stride of r. We experiment on $r = 4$ and $r = 8$ for LSUN and CelebA-HQ datasets. In the CIFAR10 experiment, we use $r = 2$ and $r = 4$. In colorization experiments, we simply degrade color images to gray by averaging images along the channels of each pixel. All tasks are evaluated on hundred samples that are sampled from the evaluation dataset. Table 6 reports the exact set of hyperparameters that we used in our experiments. We set $K = 2$ for colorization and $K = 1$ for denoising and SR tasks.

On CIFAR10 experiments, to fairly compare RGM-KLD-D and the naive version of RGM-KLD-SR, we train both models with the same degradation steps of three ($T = 3$). For RGM-KLD-D, we used $\mathbf{A}_k = e^{-\tilde{\beta}_k}\mathbf{I}$ and $\mathbf{\Sigma}_k = \left(1 - e^{-2\tilde{\beta}_k}\right)^2 \mathbf{I}$. For RGM-KLD-SR, we used $\mathbf{A}_k = e^{-\tilde{\beta}_k}\mathbf{P}_k$ and $\mathbf{\Sigma}_k = \left(1 - e^{-2\tilde{\beta}_k}\right)^2 \mathbf{P}_k^\top \mathbf{P}_k$.

Table 6: Hyperparameters used for solving inverse problems.

	CIFAR10					LSUN/CelebA-HQ		
	SR($\times$2)	SR($\times$4)	$\sigma = 10/255$	$\sigma = 20/255$	$\sigma = 40/255$	SR($\times$4)	SR($\times$8)	Color
M	5	10	10	20	10	40	40	20
λ	0.2	0.1	0.01	5	5	10	10	5
α	0.2	0.2	0.2	0.1	0.1	0.05	0.05	0.5

Baselines We employed two main comparison models, namely DDRM (Kawar et al., 2022) and GAN baseline, which is close to our work. Similar to our method, both comparison models assume that a degradation matrix is given and they iteratively update degraded images by using their knowledge obtained from the pretrained network and degradation matrix. Moreover, our model and these comparisons do not require heavy additional training. The implementation of DDRM follows its original implementation. The implementation of GAN baseline mainly follows the implementation of DGP (Pan et al., 2021), however, instead of using BigGAN (Brock et al., 2018), we replaced it with a pretrained model of StyleSwin (Zhang et al., 2022), which is one of the state-of-the-art. For discriminator loss of DGP, we used the last feature vector of StyleSwin discriminator. We additionally adjusted the weights of the losses. For experiments in SR, we use an MSE loss weight of 1.0 and a discriminator loss weight of 1.0. For colorization, we use MSE loss weight of 1.0 and discriminator loss of 1.0 for the previous 400 iterations and 0.1 after that. Other hyperparameters of GAN baseline implementation follow Pan et al. (2021).

We also compare our model with SDEdit (Meng et al., 2021), a stroke-based diffusion model. In the implementation of SDEdit, we use total denoising steps of 200 with the number of repeats of three.

C. Additional Results

C.1. Additional Ablation Studies

In this section, we include additional ablation studies on our training procedure and the forward process schedule. All experiments are conducted on the CIFAR10 dataset and focused on RGM-KLD-D.

Directly restoring the data distribution Given a k-th degraded image $\mathbf{y}_k$, the generator is trained to restore the original image in one shot. Therefore, we can train RGMs to directly restore the real image distribution from each degraded step k. RGM-KLD-D trained in this say is denoted by *Directly matching data* in Table 7. This model was trained in the same forward process as RGM-KLD-D ($T = 4$). The FID score shows that the model has difficulties in learning the data distribution, falling short of FID score by 21.2. It seems that it is still difficult to directly restore the image of the real data distribution from a severely degraded image $\mathbf{y}_k$ ($k \approx T$) even with the help of auxiliary variable $\mathbf{z}$.

Table 7: Additional ablation studies on CIFAR10 experiments.

Model	FID (↓)
Directly matching data	21.2
RGM-KLD-D w/ posterior	3.52
RGM-KLD-D ($T = 8$)	6.50
RGM-KLD-D ($T = 4$)	3.04

Training with posterior sampling While training, there are two ways to sample $\hat{y}_{k-1}$ from $\hat{y}_k$ (See line 7 of Algorithm 1 and 2). The posterior sampling (line 7 of Algorithm 1) is theoretically well-grounded since it minimizes the statistical MAP loss of the posterior distribution. However, to obtain an explicit form of posterior sampling, the forward process should be constrained to satisfy the conditions (12). Since the noising forward process of RGM-KLD-D satisfies these conditions, we trained RGM-KLD-D with both posterior sampling (Algorithm 1) and prior sampling (Algorithm 2) under the same setup. In Table 7, *RGM-KLD-D ($T = 4$)* and *RGM-KLD-D w/ posterior* refer to the model trained with prior and posterior sampling, respectively. As shown in Table 7, both models achieve similar results in terms of FID score, where RGM-KLD-D with prior sampling slightly precedes posterior sampling. This verifies that the two training objectives of Algorithms 1 and 2 are somewhat consistent. Because the performance is a bit better, we adopt the prior sampling in all our experimental studies.

Effect of the number of forward steps The number of forward steps is one of the important factors affecting the performance of the model. We investigated this in Section 4.3 by comparing a four-step model RGM-KLD-D ($T = 4$) with the RGM-KLD-D ($T = 1$), where we use only one degradation step. As reported in Table 3, RGM-KLD-D ($T = 1$) struggles to learn the data distribution because it needs to recover the real data distribution directly from Gaussian noise with one chance. On the other hand, RGM-KLD-D ($T = 4$) estimates the data density well. Besides, what happens when we use more steps? Since our RGMs learn the data distribution in a way that restores the distribution of the previous degradation step $(k - 1)$ distribution from the k-th degraded distribution, one may expect that the models will be easier to estimate the density as the distribution between the two steps is closer by dividing the forward process with more steps. However, the opposite results are presented in Table 7. The results show that RGM-KLD-D ($T = 8$) attains a higher FID score. In other words, dividing the forward process into smaller pieces does not enhance the model performance. In addition, this phenomenon is also observed for Directly matching data. RGM-KLD-D ($T = 1$) can actually be regarded as Directly matching data ($T = 1$), whereas the Directly matching data presented in the table uses $T = 4$. Comparing these two, we can observe that the model using fewer degradation steps performs better. Xiao et al. (2021a) reported a similar tendency. Choosing appropriate T is crucial for algorithmic performance, but not straightforward how many steps are optimal.

Reducing mode collapse using data fidelity Lastly, we examine the influence of the data fidelity term in our MAP-based estimation. To quantify the contribution of the fidelity term, we trained RGM-KLD-D by the loss function without the data fidelity loss (termed by RGM-KLD-D ($\lambda = \infty$)) in Section 4.3, and we reached an FID score of 32.5 (See Table 3). This result clearly motivates our objective. Moreover, we observe the mode collapse for RGM-KLD-D ($\lambda = \infty$), which is the one of common failure modes of GAN. As evidence, generated samples are presented in Figure 18. Comparing samples generated by our RGM-KLD-D (see Figure 2) to Figure 18, it is clear that images generated by RGM-KLD-D have higher diversity and better quality. The results verify that it is beneficial to train our RGMs together with the data fidelity term.

C.2. Comparison with existing models using various destruction

Recently, several works introduce various degradation processes as an alternative to the diffusion process. Rissanen et al. (2022) proposed an inverse heat dissipation model (IHDM) with a forward blurring process inspired by heat equation. Afterward, Hoogeboom & Salimans (2022) established a theoretical bridge between diffusion models and IHDM using Fourier transform. Based on this insight, they built a blurring diffusion model. Daras et al. (2022) proposed a general framework for learning the score function for any linear corruption process. Moreover, Cold Diffusion (Bansal et al., 2022) proposed a new family of models using deterministic degradation processes. Similarly, the proposed RGMs can leverage general linear degradation processes. Therefore, we compare the performance of RGMs with the aforementioned related works in Table 8. In comparison with our model itself, the change in the forward process brings FID improvement. But compared to other models, we can observe how efficiently our proposed method produces high-quality images.

Table 8: Comparison with restoration-based models with various forward processes. Sample quality on CIFAR10 is measured by FID score.

Model	FID (↓)	NFE
Cold Diffusion (SR) (Bansal et al., 2022)	152.76	3
Cold Diffusion (Blur) (Bansal et al., 2022)	80.08	50
IHDM (Rissanen et al., 2022)	18.96	200
Soft Diffusion (Daras et al., 2022)	3.86	≤ 100
Soft Diffusion (Blur) (Daras et al., 2022)	4.64	≤ 100
Blurring Diffusion (Hoogeboom & Salimans, 2022)	3.17	1000
RGM-KLD-D	3.04	4
RGM-KLD-SR	2.47	7

C.3. Additional Results on Inverse Problems

To quantify the performance of our RGM, we report signal-to-noise ratio (PSNR), which measures faithfulness to the ground-truth image. Also, as a perceptual metric, we include structural similarity index measure (SSIM) (Wang et al., 2004) that quantifies the image. Table 9 summarizes the PSNR and SSIM performances of colorization and super-resolution (SR) on CelebA-HQ and LSUN datasets. Since the primary goal of SDEdit is to generate a realistic and faithful image in the absence of paired data, we did not make a quantitative comparison with SDEdit. But we include qualitative comparisons.

Colorization The goal of image colorization is to restore a gray-scale image to a colorful image with RGB channels. We present more colorization results on CelebA-HQ and LSUN church in Figure 11 and 12, respectively. Results reported in Table 9 show that our RGM achieves comparable and sometimes even better performance than baselines. From the qualitative results, we can observe that our RGM is able to reconstruct more faithful and realistic images than other models.

Super-resolution Super-resolution aims at recovering high-resolution images corresponding to a given low-resolution image. We consider downsampled images with two scale factors 4 and 8. We also compare SR results with bicubic interpolation. Figure 13 and 14 present the qualitative comparisons. Compared against bicubic upsampling, bicubic attains

Table 9: Colorization and super-resolution results of different methods.

Model	Colorization				Super-Resolution							
	LSUN		CelebA-HQ		LSUN				CelebA-HQ			
					(×4)		(×8)		(×4)		(×8)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
RGM	23.78	0.93	25.57	0.93	22.74	0.65	19.96	0.48	28.51	0.81	24.86	0.70
DDRM	23.68	0.94	23.94	0.93	23.22	0.67	20.61	0.51	29.32	0.83	26.23	0.73
GAN baseline	20.02	0.81	24.79	0.88	20.32	0.48	18.06	0.34	26.77	0.71	23.92	0.59

higher PSNR and SSIM values. However, we can observe from Figure 13 and 14 that bicubic interpolation results in blurry images, and RGM super-resolves more plausible images. Also, visual differences between RGM and DDRM are qualitatively not large.

C.4. Additional results of varying $\mathbf{z}$

We investigated the influence of the auxiliary variable $\mathbf{z}$ in Section 4.3. Here, we include more observations in Figure 10.

C.5. Additional Qualitative Results on Generation

We present more generated image samples in Figures 15, 16, 17, 19, and 20.

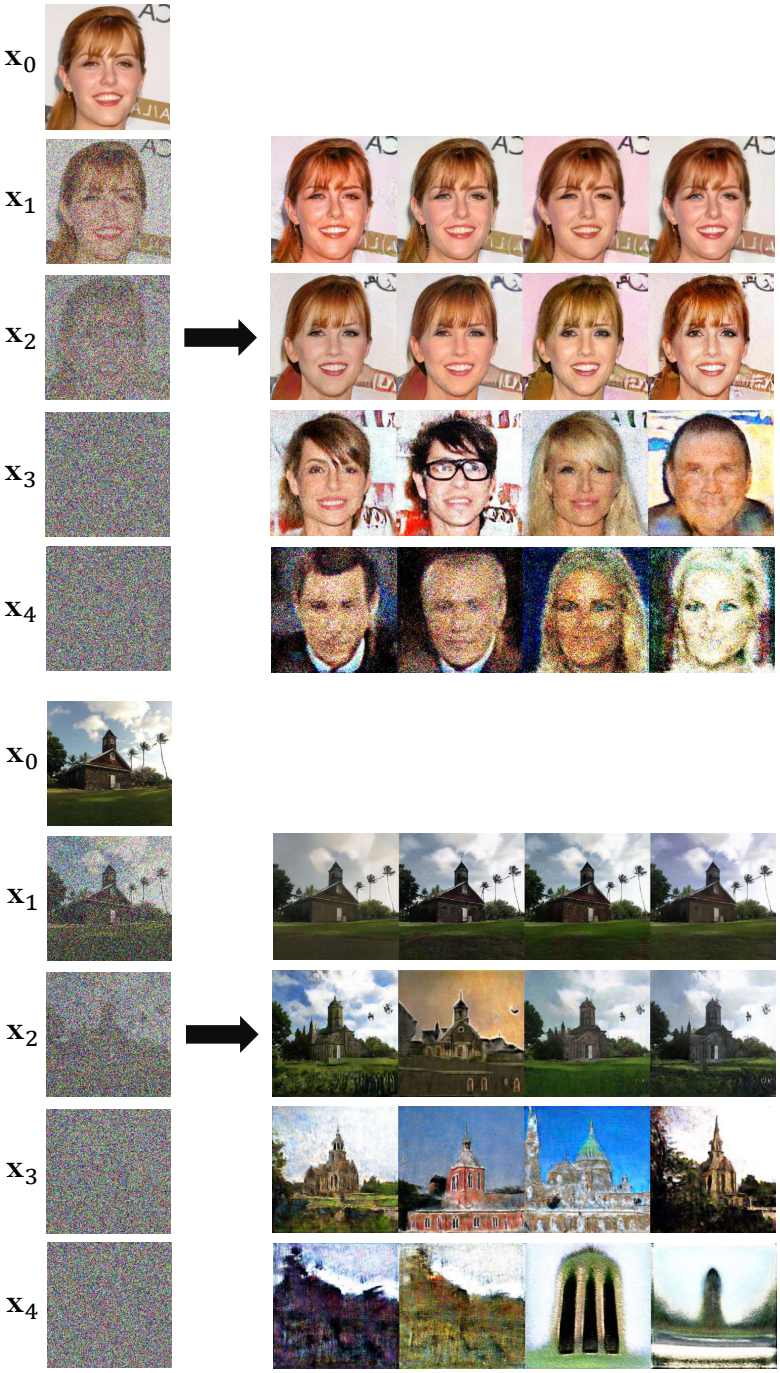

Figure 10: Illustration of the effect of varying $\mathbf{z}$ on CelebA-HQ (top) and LSUN (bottom). The images in the leftmost column depict the selected trajectory $\{\mathbf{x}_k\}_{k=1}^4$ degraded from an image $\mathbf{x}_0$. Each row on the right presents restored images of $\mathbf{x}_t$ using four different random auxiliary values $\mathbf{z}$. When the noise level is small, they generate almost identical images, which means that the restoration problem is almost well-posed. As the noise level increases, however, each degraded observation $\mathbf{x}_k$ estimates diverse images depending on the $\mathbf{z}$. In other words, the larger the noise, the more severe the ill-posedness, and the results validate that a much wider restoration is possible through the introduction of $\mathbf{z}$.

Figure 11: **Colorization**. Qualitative comparison on CelebA-HQ.

Figure 12: **Colorization**. Qualitative comparison on LSUN church.

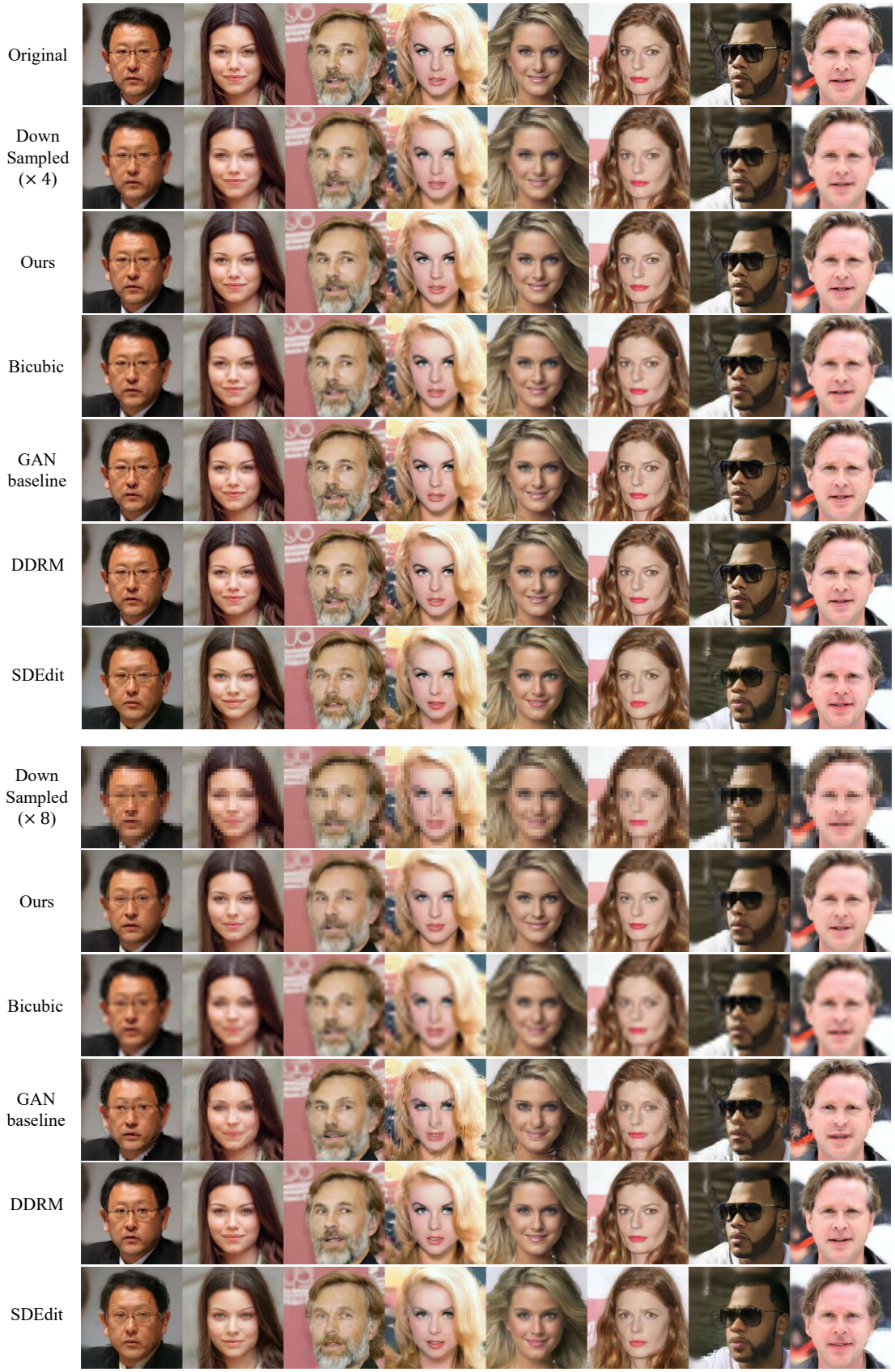

Figure 13: **Super-resolution**. Qualitative comparison on CelebA-HQ.

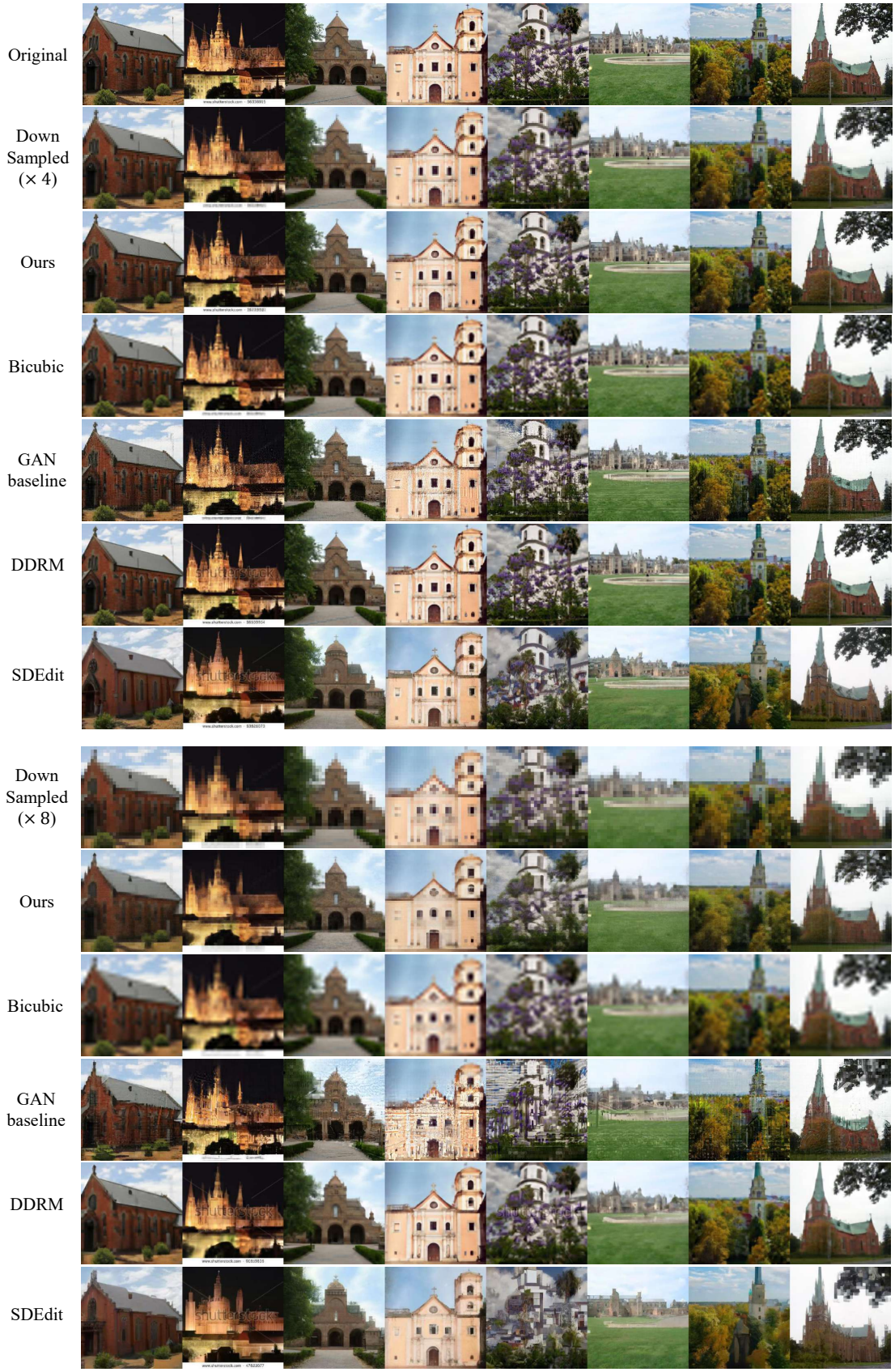

Figure 14: **Super-resolution**. Qualitative comparison on LSUN church.

Figure 15: Generated samples of RGM-DSWD-D on CIFAR10.

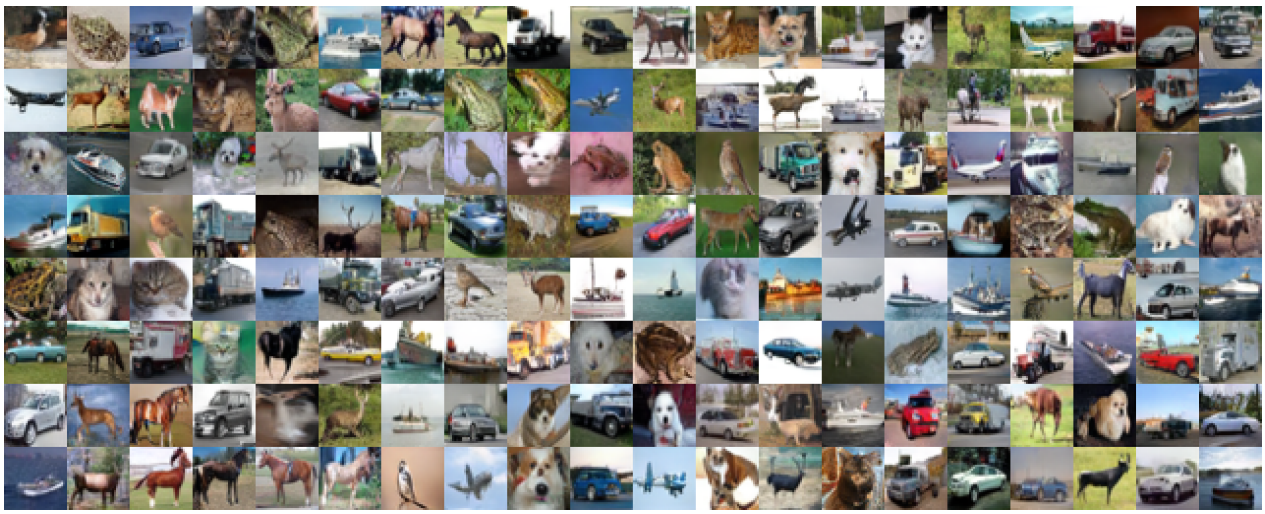

Figure 16: Generated samples of RGM-KLD-D on CIFAR10.

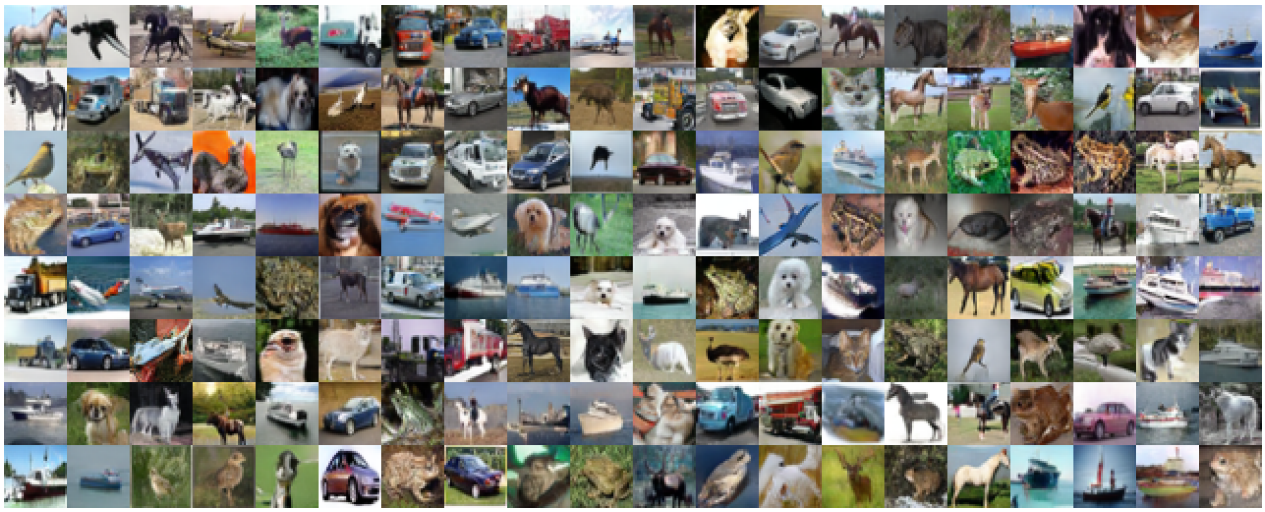

Figure 17: Generated samples of RGM-KLD-SR on CIFAR10.

Figure 18: Mode collapse of RGM-KLD-D trained without the data fidelity term. Sampled images of RGM-KLD-D ($\lambda = \infty$) seem repetitive.

Figure 19: Additional qualitative results of RGM-KLD-D trained on CelebA-HQ-256.

Figure 20: More qualitative results of RGM-KLD-D trained on LSUN Church.