Reconsidering Noise for Denoising Diffusion Probabilistic Models

Stephen D. Liang*
Hewlett Packard Enterprise
San Jose, CA 95002, USA
stephendliang@gmail.com

Abstract

Denoising Diffusion Probabilistic Models (DDPMs) typically utilize white Gaussian noise in their processes. In this paper, we explore several theoretical aspects of noise in DDPMs. We derive a necessary condition for the input of the forward diffusion process to match the denoised output, as well as a sufficient condition for when they differ. Our findings show that minimizing the Mean Square Error (MSE) between the actual and predicted noise in a DDPM is more effective with colored Gaussian noise than with white Gaussian noise, and that non-Gaussian noise offers further improvements in MSE minimization. Additionally, we demonstrate that the probability of error between the input and denoised output in a DDPM is reduced when using colored Gaussian noise compared to white Gaussian noise. Furthermore, we show that a DDPM trained with white Gaussian noise can effectively denoise processes involving any zero-mean symmetric distribution noise. Theoretical results are validated through experiments using the Hugging Face Hub 1000 butterfly pictures dataset and the LSUN Church-256 dataset, with experimental outcomes confirming our theoretical findings.

1 Introduction

Denoising Diffusion Probabilistic Models (DDPMs) have emerged as a powerful class of generative models, recognized for their ability to produce high-quality samples. Unlike traditional generative models such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), DDPMs are rooted in the principles of diffusion processes from statistical physics [37][15]. The fundamental concept of DDPMs is to capture the data distribution by inverting a diffusion process that incrementally adds noise to the data. This method comprises two primary phases: the forward process and the reverse process.

During the forward diffusion process, a data point is gradually corrupted by Gaussian noise over a series of time steps, resulting in progressively noisier samples. The reverse denoising process, which constitutes the generative phase of DDPMs, seeks to recover the original data from these noisy observations. This process is parameterized by a neural network that learns to denoise the samples iteratively, effectively reversing the diffusion process. The network is trained by minimizing a variational bound on the negative log-likelihood of the data, which involves reducing the Kullback-Leibler (KL) divergence between the true and approximate posterior distributions [15].

In most current implementations of DDPMs, white Gaussian noise is employed because it's popular and easy to make analysis. This raises questions: Can colored Gaussian noise or non-Gaussian noise be used instead? Would the performance in terms of Mean Square Error (MSE) improve with these alternative noise types? What are the theoretical guidelines to consider different noises in DDPM? This paper investigates these fundamental theoretical questions.

*This work was done while the author was employed by Hewlett Packard Enterprise.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: SPIGM.

The contributions of this paper are as follows:

1. We analyze the probability lower bound for the scenario where the input to the forward diffusion process matches the denoised output, deriving a necessary condition for this match. We also establish a sufficient condition for when they differ.
2. To train the noise prediction network in a DDPM, it is essential to minimize the MSE between the actual noise and the predicted noise. We demonstrate that the MSE is smaller when the noise is colored Gaussian noise compared to white Gaussian noise.
3. We show that non-Gaussian noise outperforms Gaussian noise in minimizing the MSE between the actual noise and the predicted noise in a DDPM if they have the same covariance matrix.
4. We prove that the probability of error between the input and denoised output of a DDPM has a smaller lower bound when using colored Gaussian noise compared to white Gaussian noise.
5. We demonstrate that a DDPM trained with white Gaussian noise can be effectively used in the denoising process when the noise samples follow any zero-mean symmetric distribution.

2 Related Work

DDPM and Modifications: In [31], several modifications to diffusion models were proposed, including improvements to sampling speed and log-likelihood, such as enhanced noise scheduling and reduced gradient noise. A Discrete Denoising Diffusion Probabilistic Model (D3PM) was introduced for discrete data, generalizing the multinomial diffusion model [2]. In [7], an Iterative Latent Variable Refinement (ILVR) method was proposed for DDPMs to generate high-quality images using a reference image as guidance. In [49], multimodal image fusion was explored using DDPMs, where the problem was divided into an unconditional generation subproblem and a maximum likelihood subproblem. A pyramidal DDPM capable of generating high-resolution images from much coarser resolution images was proposed in [35], utilizing a single score function trained with positional embedding. Recently, a multi-task denoising diffusion framework, DiffusionMTL, was introduced in [46]. It integrates a joint diffusion and denoising paradigm to model potential noisy distributions in task prediction or feature maps, generating refined outputs for different tasks.

Other Denoising Diffusion Models: A Denoising Diffusion Implicit Model (DDIM) was proposed in [38], introducing a class of non-Markovian diffusion processes with the same training objective but a much faster reverse denoising process. In [48], a generalized DDIM was proposed by examining the mechanism of DDIM from a numerical perspective. A Denoising Diffusion Restoration Model (DDRM) was introduced to leverage a pre-trained denoising diffusion generative model for solving any linear inverse problem [20]. In [22], SinDDM was presented as a method to train a DDM using a single image by learning the internal statistics of the training image through a multi-scale diffusion process. A Bilateral Denoising Diffusion Model (BDDM), which requires significantly fewer steps to generate high-quality samples, was proposed in [24]. A Denoising Diffusion Bridge Model (DDBM) was introduced based on diffusion bridges [50], where the score of the diffusion bridge was learned from data to map between two endpoint distributions. In [13], a Semi-Implicit Denoising Diffusion Model (SIDDM) was proposed by matching implicit and explicit factors, where an implicit model was used to align the noise data marginal distributions with the forward diffusion explicit conditional distribution. To enable diffusion model training with limited computational resources while maintaining quality and flexibility, a Latent Diffusion Model (LDM) was proposed using the latent space of powerful pretrained autoencoders [33]. Stable Diffusion [6, 43, 11], based on LDM, employs U-Net [34] and transformer-based blocks with a self-attention mechanism [44]. Stable Diffusion is a text-conditioned LDM leveraging Contrastive Language–Image Pre-training (CLIP) [32, 30, 25]. CLIP connects text with images, and its text encoder transforms text into numerical representations for stable diffusion [40, 1].

Some Theoretical Foundations of DDPM: In [21], the variational lower bound (VLB) of diffusion models was simplified to a function of the signal-to-noise ratio of the diffused data. Sampling from DPMs was studied in [26], where an exact solution to the diffusion ordinary differential equations (ODEs) was proposed. A Higher-Order Denoising Diffusion Solver (GENIE), based on truncated Taylor methods, was introduced to accelerate synthesis using higher-order gradients of the perturbed

data distribution [10]. To minimize the cumulative estimation gap between predicted and actual trajectories, a sequence-aware loss was proposed to improve sampling quality [29]. In [4], an analytic diffusion probabilistic model was proposed, which does not require training and can estimate variance and KL divergence. This work was extended to the scenario with imperfect means for optimal covariance estimation in diffusion probabilistic models [3].

Noise Reconsideration: In DDPMs, white Gaussian noise has traditionally been used for both the diffusion and denoising processes [15, 8]. White noise is characterized by a uniform power spectral density (PSD) across all frequencies [23]. In [28], Gamma noise and a mixture of Gaussian noise were employed in the diffusion process, leading to improved DDPM performance. Noise level estimation and noise scheduling adjustments were performed for DDPM in [36]. In [41], the removal of structured noise in diffusion models was studied. A multi-scale simplex noise diffusion process was proposed for anomaly detection in [45], where it was found that simplex noise significantly outperformed Gaussian diffusion. Recently, blue noise, characterized by a PSD that increases with frequency (indicating higher power at higher frequencies), was used in diffusion models [16]. This time-varying noise model incorporates correlated noise into the training process, resulting in better image quality than white Gaussian noise in DDPM [16]. However, these approaches have been primarily experimental, with no accompanying theoretical analysis. In this paper, we focus on theoretical studies regarding noise in DDPMs.

3 Background

The forward diffusion process and the reverse denoising process in DDPMs can be mathematically described as follows.

3.1 Forward Diffusion Process

The forward diffusion process incrementally adds noise to the data over a sequence of T timesteps, transforming an initial data sample $\mathbf{x}_0$ into a noisy sample $\mathbf{x}_T$. This process is represented as a Markov chain, with Gaussian noise added at each step.

Given an initial data sample $\mathbf{x}_0 \sim q(\mathbf{x}_0)$, the forward process is defined as [15]:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}), \tag{1}$$

where β_t is a variance schedule that controls the amount of noise added at each timestep t. The full forward process can be expressed as [15]:

$$q(\mathbf{x}_{1:T} \mid \mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t \mid \mathbf{x}_{t-1}). \tag{2}$$

The marginal distribution of $\mathbf{x}_t$ given $\mathbf{x}_0$ can be derived as [15]:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}), \tag{3}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$. The noise scheduler β_s is designed such that at the end of the diffusion process $\bar{\alpha}_t \to 0$, leading to [15]:

$$q(\mathbf{x}_T) \approx \mathcal{N}(\mathbf{x}_t; \mathbf{0}, \mathbf{I}), \tag{4}$$

3.2 Reverse Denoising Process

The reverse denoising process seeks to reconstruct the original data sample $\mathbf{x}_0$ from the noisy sample $\mathbf{x}_T$ by iteratively removing noise. This process is parameterized by a neural network that approximates the reverse conditional distributions.

The reverse process is defined as [15]:

$$p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)), \tag{5}$$

where μ_θ and Σ_θ are parameters learned by the neural network. The full reverse process is expressed as [15]:

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t), \tag{6}$$

where

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_t; \mathbf{0}, \mathbf{I}). \tag{7}$$

The model is trained by minimizing the variational bound on the negative log-likelihood of the data. The training objective can be written as [15]:

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{x}_{0:T})} \left[\sum_{t=1}^{T} D_{KL}(q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)) \right], \tag{8}$$

where D_{KL} represents the Kullback-Leibler divergence between the true posterior $q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0)$ and the model distribution $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$.

Denoising Diffusion Models provide a robust framework for generative modeling by leveraging an iterative denoising process. By training a neural network to approximate the reverse process, these models can generate high-quality samples from complex data distributions.

4 Theoretical Studies on Noise for DDPMs

4.1 A Necessary Condition for Denoised Output Matching Input

For ease of analysis, let us denote the output from the reverse denoising process as $\hat{\mathbf{x}}_0$. In most cases, the denoised output $\hat{\mathbf{x}}_0$ differs from the initial input $\mathbf{x}_0$. However, understanding the necessary condition for $\hat{\mathbf{x}}_0$ to match $\mathbf{x}_0$ is a fundamental question we address before investigating the noise.

Our theoretical findings regarding the condition for $\hat{\mathbf{x}}_0$ to match $\mathbf{x}_0$ are encapsulated in the following theorem.

Theorem 1. *For a DDPM with input $\mathbf{x}_0$ and denoised output $\hat{\mathbf{x}}_0$, the probability of error, denoted as $p(e) = Pr(\|\mathbf{x}_0 - \hat{\mathbf{x}}_0\| \geq \epsilon)$ where ϵ is a small number, is lower bounded by:*

$$p(e) \geq \frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N |\mathbf{K}_{\mathbf{x_0}}|} + 1, \tag{9}$$

where a necessary condition for the DDPM output $\hat{\mathbf{x}}_0$ to match $\mathbf{x}_0$ is:

$$\frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N |\mathbf{K}_{\mathbf{x_0}}|} + 1 \leq 0, \tag{10}$$

with N representing the length of $\mathbf{x}_0$ after vectorization, and $\mathbf{K}_{\mathbf{x_0}}$ being the covariance matrix of $\mathbf{x}_0$.

The proof of Theorem 1 is provided in the Appendix.

Corollary 1. *For a DDPM with input $\mathbf{x}_0$ and denoised output $\hat{\mathbf{x}}_0$, it is impossible for $\mathbf{x}_0$ and $\hat{\mathbf{x}}_0$ to be identical if:*

$$|\mathbf{K}_{\mathbf{x_0}}| < \frac{1}{(2\pi e)^N}. \tag{11}$$

The proof of Corollary 1 is provided in Appendix.

4.2 Reconsideration of Noise in DDPM

Blue noise, as utilized in [16], is a specific type of colored noise. In this paper, we explore a more general case by studying the use of colored Gaussian noise in DDPMs. Colored Gaussian noise has a spectral density that varies with frequency, meaning that certain frequency components are more prominent than others [27]. When using colored Gaussian noise, the forward diffusion process can be defined as:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{12}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1 - \beta_s)$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_\mathbf{n})$. The forward process is given by:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{K}_\mathbf{n}), \tag{13}$$

where $\mathbf{K}_\mathbf{n}$ is the covariance matrix of the colored Gaussian noise. Unlike white noise, where the covariance matrix is diagonal (indicating uncorrelated noise components), the covariance matrix of colored noise is nondiagonal, reflecting the correlation among noise components [23].

The marginal distribution of $\mathbf{x}_t$ given $\mathbf{x}_0$ can be expressed as:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{K_n}), \tag{14}$$

and at the end of the diffusion process:

$$q(\mathbf{x}_T) \approx \mathcal{N}(\mathbf{x}_t; \mathbf{0}, \mathbf{K_n}). \tag{15}$$

The denoising diffusion process aims to maximize the evidence lower bound (ELBO), which is equivalent to minimizing the following KL divergence [3]:

$$\min_{\mu_n, \mathbf{K}_n} D_{KL}(q(x_{0:N}) \parallel p(x_{0:N})), \tag{16}$$

The optimal values of μ_n and $\mathbf{K}_n$ in analytical forms were determined for DDPMs in [4], and variance for imperfect means was studied in [3]. In [15], the optimal μ_n was estimated using a noise prediction network $\hat{\boldsymbol{\epsilon}}_t$, obtained by minimizing the Mean Square Error (MSE) [3]:

$$\min_{\hat{\mu}_t, t=1,2,\ldots,T} \mathbb{E}_t \mathbb{E}_{q(x_0,x_t)} \left\| \boldsymbol{\epsilon}_t - \hat{\boldsymbol{\epsilon}}_t \right\|_2^2. \tag{17}$$

It was shown that the estimated mean value of noise $\mathbf{x}_n$ is [15]:

$$\hat{\boldsymbol{\mu}}_n(\mathbf{x}_t) = \tilde{\boldsymbol{\mu}}_n(\mathbf{x}_t, \frac{1}{\sqrt{\bar{\alpha}_n}}(\mathbf{x}_t - \sqrt{\bar{\beta}_t}\hat{\boldsymbol{\epsilon}}_n(\mathbf{x}_t))). \tag{18}$$

For white Gaussian noise with estimated mean values in (18), the optimal covariance matrix satisfying (16) is a diagonal matrix with diagonal values [4, 3]:

$$\tilde{\boldsymbol{\sigma}}_n^*(\mathbf{x}_t)^2 = \lambda_t^2 \mathbf{1} + \gamma_t^2 \frac{\bar{\beta}_t}{\bar{\alpha}_t} \mathbb{E}_{q(x_0|x_n)}[(\boldsymbol{\epsilon}_n - \hat{\boldsymbol{\epsilon}}_n(\mathbf{x}_t))^2], \tag{19}$$

which is a vector of length N. Similarly, for colored Gaussian noise with mean values in (18), the optimal covariance matrix satisfying (16) is [3]:

$$\mathbf{K}_\mathbf{n}^*(\mathbf{x}_t) = \lambda_t^2 \mathbf{I} + \gamma_t^2 \frac{\bar{\beta}_t}{\bar{\alpha}_t} \mathbb{E}_{q(x_0|x_n)}[(\boldsymbol{\epsilon}_n - \hat{\boldsymbol{\epsilon}}_n(\mathbf{x}_t))(\boldsymbol{\epsilon}_n - \hat{\boldsymbol{\epsilon}}_n(\mathbf{x}_t))^T]. \tag{20}$$

Based on these theoretical results, we can demonstrate that colored Gaussian noise is superior to white Gaussian noise in DDPMs in terms of MSE and probability of error. We come up with the following Theorems and Corollaries, and their proofs are provided in the Appendix.

Theorem 2. *For a DDPM, to obtain the noise prediction network $\hat{\boldsymbol{\epsilon}}_n$, the MSE in (17) must be minimized. The MSE is smaller when the noise $\boldsymbol{\epsilon}_t$ is colored Gaussian noise compared to white Gaussian noise.*

Corollary 2. *For a DDPM, non-Gaussian noise performs better than Gaussian noise in minimizing the MSE in (17) if they have the same covariance matrix.*

Theorem 3. *For a DDPM with $\mathbf{x}_t$ in the forward diffusion process and $\hat{\mathbf{x}}_t$ in the reverse denoising process, the probability of error at timestep t, denoted as $p_t(e) = Pr(\mathbf{x}_t \neq \hat{\mathbf{x}}_t)$, is lower bounded by:*

$$p_t(e) \geq \frac{\log(1-\bar{\alpha}_T) - 2}{\log(2\pi e)^N |\mathbf{K_{x_0}}|} + 1, \tag{21}$$

with N representing the length of $\mathbf{x}_0$ after vectorization, and $\mathbf{K_{x_0}}$ being the covariance matrix of $\mathbf{x}_0$.

Theorem 4. *For a DDPM with $\mathbf{x}_t$ in the forward diffusion process and $\hat{\mathbf{x}}_t$ in the reverse denoising process, the probability of error at timestep t, denoted as $p_t(e) = Pr(\mathbf{x}_t \neq \hat{\mathbf{x}}_t)$, has a smaller lower bound with colored Gaussian noise than with white Gaussian noise.*

Corollary 3. *For a DDPM, it is possible for $\mathbf{x}_t$ and $\hat{\mathbf{x}}_t$ to match each other when $t = 0$.*

A DDPM trained with white Gaussian noise can effectively perform the denoising diffusion process even if the noise samples in $\mathbf{x}_T$ follow any symmetric distribution with a zero mean. We provide the following theorem.

Theorem 5. *Let a DDPM (variance-preserving (VP) diffusion) be trained with a Gaussian forward process $q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\big(\sqrt{\bar{\alpha}_t}\,\mathbf{x}_0,\, (1-\bar{\alpha}_t)\mathbf{I}\big)$ and let its score network be exact, i.e. $s_\theta(\mathbf{x}, t) = \nabla_\mathbf{x} \log q_t(\mathbf{x})$. Consider sampling with the* probability flow Ordinary Differential Equation (ODE) *associated with the VP Stochastic Differential Equation (SDE). If the initial sample $\mathbf{x}_T$ is drawn from* any *zero-mean symmetric distribution r_T with a density and finite second moment, then the reverse-time dynamics $\{\mathbf{x}_t\}_{t\downarrow 0}$ produced by the ODE* effectively performs denoising diffusion *in the sense that the KL divergence between the law μ_t of $\mathbf{x}_t$ and the Gaussian-corrupted data marginal q_t is nonincreasing in t and equals 0 at $t = 0$. In particular, the terminal law r_T need not be Gaussian.*

5 Experiments

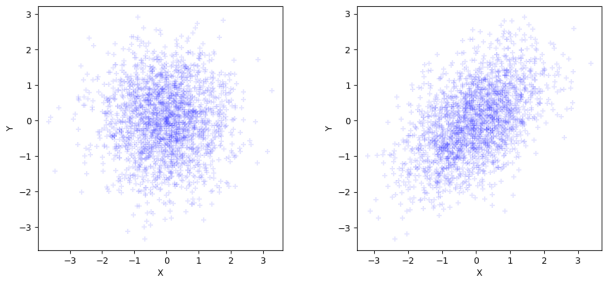
We conducted experiments using two datasets. The first dataset consists of 1000 butterfly images from the Hugging Face Hub [18], with each image having a resolution of $3 \times 32 \times 32$. This dataset was used to test scenarios where the noise in both the forward diffusion and reverse denoising processes follows the same statistical distribution, such as when both are uniform noise. The second dataset used is the LSUN-Church dataset [47], which has images with a resolution of $3 \times 256 \times 256$. This dataset was employed to explore cases where the forward diffusion process uses white Gaussian noise, while the noise samples in the reverse denoising process might follow different distributions, such as Laplacian noise.

Our DDPM followed the design described in [15]. We conducted experiments to validate our theoretical findings using white Gaussian, colored Gaussian, and non-Gaussian noises. A 2D UNet architecture was employed for the DDPM [34].

In Fig. 1, we illustrate two examples of zero-mean Gaussian noise with different covariance matrices $\mathbf{K_n}$. Despite having different covariance structures, all noise types share the same average power:

$$P = \text{tr}(\mathbf{K_n}), \tag{22}$$

since their traces are identical. Fig. 1(a) represents white Gaussian noise with a unit covariance matrix, while Fig. 1(b) shows colored Gaussian noise.



(a) (b)

Figure 1: Illustration of Gaussian noises. (a) White Gaussian noise with zero-mean and $\mathbf{K_n} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, (b) Colored Gaussian noise with zero-mean and $\mathbf{K_n} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$.

As shown in Fig. 1, white Gaussian noise (Fig. 1a) has no directional preference, while colored Gaussian noise (Fig. 1b) exhibits directional preferences. The colored noise has stronger power (larger eigenvalue of $\mathbf{K_n}$, 1.5) in the direction $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ compared to the weaker direction $\begin{bmatrix} -1 \\ 1 \end{bmatrix}$, which has a smaller eigenvalue of $\mathbf{K_n}$, 0.5. We also used non-Gaussian noise, specifically uniform noise, which also has a power of 1 in each dimension (details in the supplementary material). These noises were used in our experiments on both datasets.

5.1 Experiments with Butterfly Pictures Dataset

For this dataset, the training noise and the noise used in the denoising process share the same statistical distributions. Our 2D-UNet encoder consists of four downsample blocks, and the decoder has four upsample blocks. Each UNet block contains two ResNet layers. In Fig. 2, we summarize the MSE versus the number of epochs during 2D-UNet training. As shown, colored Gaussian noise results in lower MSE than white Gaussian noise, which aligns with our theoretical result in Theorem 2.

We present the generated outputs from the DDPM for the three different noises in Fig. 3. We used $T = 1000$ in our experiments and provided one set of examples for each noise in the denoising diffusion process from $t = 300$ to $t = 0$. As shown, all three types of noise were able to produce high-quality butterfly images after denoising.

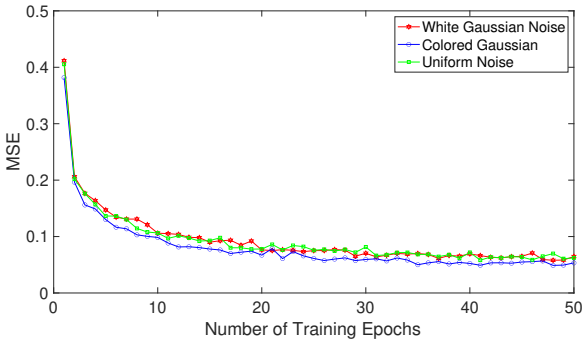


Figure 2: MSE versus the number of training epochs for different noises.

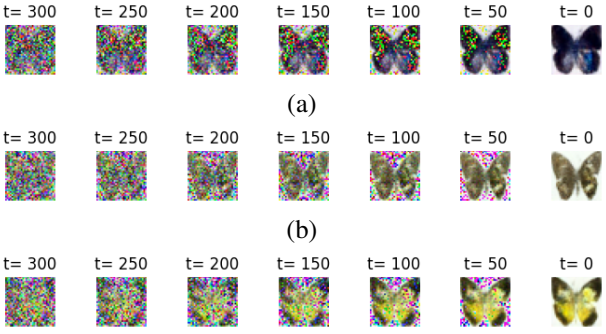


Figure 3: The reverse denoising process of DDPM at different times t for different zero-mean noises. (a) White Gaussian noise, (b) Colored Gaussian noise, (c) Uniform noise.

We used the Fréchet Inception Distance (FID) to evaluate the similarity of generated images to real ones. The FID is defined as [14]:

$$\text{FID} = \|\mu_r - \mu_g\|_2^2 + \text{tr}\left(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r \Sigma_g}\right), \tag{23}$$

where μ_r and μ_g are the mean vectors of the real and generated data features, respectively, and Σ_r and Σ_g are the covariance matrices of the real and generated data features, respectively. $\|\cdot\|_2$ denotes the Euclidean distance. The FID evaluation was implemented using the Inception-v3 network [42], as described in [39]. In Table 1, we summarize the FID values for the three image datasets generated by the three different noises. For each noise case, we generated 100 images to evaluate the FID between the generated images dataset and the Hugging Face 1000 butterfly images.

Table 1: FID values for DDPMs with different noises.

White Gaussian	Colored Gaussian	Uniform
104.41	69.89	78.49

A lower FID indicates that the generated images are closer to real images. As observed, both colored Gaussian and uniform noise achieved better performance than white Gaussian noise, with the images generated by colored Gaussian noise having the lowest FID value.

The above experiments were performed using Google Colab with T4 GPU. The total running time was around 250 minutes.

5.2 Experiments with LSUN Church-256 Dataset

For this dataset, the DDPM was pre-trained using zero-mean white Gaussian noise, but the noise samples used in the reverse denoising process may follow different statistical distributions.

A simple unconditional image generation model, UNet2DModel, was pre-trained on the LSUN church images dataset, specifically using the google/ddpm-church-256 [17]. The UNet2DModel's encoder consists of 6 down-sampling 2D blocks, while its decoder comprises 6 up-sampling 2D blocks, with each block containing 2 ResNet layers. The model remains fixed during the denoising process.

For illustration, we plotted an output generated by the DDPM using zero-mean white and colored Gaussian noise in Fig. 4. By varying the noise seeds, we could generate any number of outputs. Notably, high-quality images were consistently produced.

(a) (b)

Figure 4: Generated outputs of DDPM with Gaussian noises. (a) White Gaussian noise, (b) Colored Gaussian noise.

Additional experiments are provided in the Appendix.

6 Conclusions

Current DDPM research predominantly relies on white Gaussian noise. We have proposed a reconsideration of the types of noise used in DDPMs, both during the training of the noise estimation network and in the denoising process of a trained DDPM. Although some previous works have experimented with non-Gaussian noises, there has been a lack of theoretical investigation.

In our work, we have proposed and analyzed the probability lower bound for scenarios where the input to the forward diffusion process aligns with the denoised output, deriving a necessary condition for this alignment.

To train the noise prediction network in a DDPM, it is crucial to minimize the MSE between the actual noise and the predicted noise. We have shown that the MSE is reduced when using colored Gaussian noise as opposed to white Gaussian noise. Furthermore, we have demonstrated that non-Gaussian noise outperforms Gaussian noise in minimizing the MSE between actual and predicted noise in a DDPM if they have the same covariance matrix.

We have proved and verified that the probability of error between the input and denoised output in a DDPM has a lower bound that is smaller when using colored Gaussian noise compared to white Gaussian noise. Moreover, we have shown that a DDPM trained with white Gaussian noise can be effectively applied to denoise processes involving any zero-mean symmetric distribution noise. Our theoretical findings have been validated using two Gaussian and six non-Gaussian noise distributions.

In summary, colored Gaussian noise is superior to white Gaussian noise, and non-Gaussian noise outperforms Gaussian noise in a DDPM. Additionally, any zero-mean symmetric distribution noise can generate a denoised image using a trained DDPM by zero-mean white Gaussian noise. This helps DDPMs with applications where noise may have uncertain distributions.

References

[1] Reza Abbasi, Mohammad Hossein Rohban, and Mahdieh Soleymani Baghshah. Deciphering the role of representation disentanglement: Investigating compositional generalization in clip models. *arXiv preprint arXiv:2407.05897*, 2024.

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

[3] Fan Bao, Chongxuan Li, Jiacheng Sun, Jun Zhu, and Bo Zhang. Estimating the optimal covariance with imperfect mean in diffusion probabilistic models. *arXiv preprint arXiv:2206.07309*, 2022.

[4] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022.

[5] Shane Barratt. A matrix gaussian distribution. *arXiv preprint arXiv:1804.11010*, 2018.

[6] Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4599–4603, 2023.

[7] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021.

[8] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11472–11481, 2022.

[9] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

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

[11] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22466–22477, 2023.

[12] Sebastien Gerchinovitz, Pierre Ménard, and Gilles Stoltz. Fano's inequality for random variables. *Statistical Science*, 35(2):178 – 201, 2020.

[13] Mingming Gong, Shaoan Xie, Wei Wei, Matthias Grundmann, Kayhan Batmanghelich, Tingbo Hou, et al. Semi-implicit denoising diffusion models (siddms). *Advances in Neural Information Processing Systems*, 36, 2024.

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

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[16] Xingchang Huang, Corentin Salaun, Cristina Vasconcelos, Christian Theobalt, Cengiz Oztireli, and Gurprit Singh. Blue noise for diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[17] HuggingFace. Ddpm church 256. `https://huggingface.co/google/ddpm-church-256`, 2022. Accessed: 2024-08-14.

[18] HuggingFace. smithsonian_butterflies_subset. `https://huggingface.co/datasets/huggan/smithsonian_butterflies_subset`, 2023. Accessed: 2024-08-13.

[19] Richard A Johnson, Irwin Miller, and John E Freund. *Probability and statistics for engineers*, volume 2000. Pearson Education London, 2000.

[20] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606, 2022.

[21] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

[22] Vladimir Kulikov, Shahar Yadin, Matan Kleiner, and Tomer Michaeli. Sinddm: A single image denoising diffusion model. In *International conference on machine learning*, pages 17920–17930. PMLR, 2023.

[23] Hui-Hsiung Kuo. *White noise distribution theory*. CRC press, 2018.

[24] Max WY Lam, Jun Wang, Rongjie Huang, Dan Su, and Dong Yu. Bilateral denoising diffusion models. *arXiv preprint arXiv:2108.11514*, 2021.

[25] Manling Li, Ruochen Xu, Shuohang Wang, Luowei Zhou, Xudong Lin, Chenguang Zhu, Michael Zeng, Heng Ji, and Shih-Fu Chang. Clip-event: Connecting text and images with event structures. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16420–16429, 2022.

[26] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

[27] Jerzy Łuczka. Non-markovian stochastic processes: Colored noise. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(2), 2005.

[28] Eliya Nachmani, Robin San Roman, and Lior Wolf. Non gaussian denoising diffusion models. *arXiv preprint arXiv:2106.07582*, 2021.

[29] Viet Nguyen, Giang Vu, Tung Nguyen Thanh, Khoat Than, and Toan Tran. On inference stability for diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14449–14456, 2024.

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

[31] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

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

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[35] Dohoon Ryu and Jong Chul Ye. Pyramidal denoising diffusion probabilistic models. *arXiv preprint arXiv:2208.01864*, 2022.

[36] Robin San-Roman, Eliya Nachmani, and Lior Wolf. Noise estimation for generative diffusion models. *arXiv preprint arXiv:2104.02600*, 2021.

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

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

[39] George Stein, Jesse Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L Caterini, Eric Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[40] Kyle Steinfeld. Clever little tricks: a socio-technical history of text-to-image generative models. *International Journal of Architectural Computing*, 21(2):211–241, 2023.

[41] Tristan SW Stevens, Hans van Gorp, Faik C Meral, Junseob Shin, Jason Yu, Jean-Luc Robert, and Ruud JG van Sloun. Removing structured noise with diffusion models. *arXiv preprint arXiv:2302.05290*, 2023.

[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[43] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*, 2022.

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

[45] Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 650–656, 2022.

[46] Hanrong Ye and Dan Xu. Diffusionmtl: Learning multi-task denoising diffusion model from partially annotated data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27960–27969, 2024.

[47] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[48] Qinsheng Zhang, Molei Tao, and Yongxin Chen. gddim: Generalized denoising diffusion implicit models. *arXiv preprint arXiv:2206.05564*, 2022.

[49] Zixiang Zhao, Haowen Bai, Yuanzhi Zhu, Jiangshe Zhang, Shuang Xu, Yulun Zhang, Kai Zhang, Deyu Meng, Radu Timofte, and Luc Van Gool. Ddfm: denoising diffusion model for multi-modality image fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8082–8093, 2023.

[50] Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. *arXiv preprint arXiv:2309.16948*, 2023.

A Proof of Theorem 1

Proof. For a DDPM, the initial data sample $\mathbf{x}_0$, the final diffusion outcome $\mathbf{x}_T$, and the denoised estimate $\hat{\mathbf{x}}_0$ form a Markov chain:

$$\mathbf{x}_0 \to \mathbf{x}_T \to \hat{\mathbf{x}}_0. \tag{24}$$

The input $\mathbf{x}_0$ to the diffusion process could be either a matrix or a vector. If it is a matrix, we can vectorize it [5], and assume the vector length is N.

For the continuous variables, define the error e as:

$$e = \begin{cases} 1 & \text{if } \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\| \geq \epsilon, \\ 0 & \text{if } \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\| < \epsilon. \end{cases} \tag{25}$$

where ϵ is a very small number. Fano's inequality can be used for discrete variables as well as continuous variables. In [9], Fano's inequality was applied to Gaussian waveform channel for channel capacity analysis, which is a continuous variable case application. Based on Fano's inequality [9, 12]:

$$p(e) \geq \frac{h(\mathbf{x}_0 \mid \mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{26}$$

$$= \frac{h(\mathbf{x}_0, \mathbf{x}_T) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{27}$$

$$= \frac{h(\mathbf{x}_T \mid \mathbf{x}_0) + h(\mathbf{x}_0) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{28}$$

$$= \frac{h(\mathbf{x}_T \mid \mathbf{x}_0) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} + 1. \tag{29}$$

Based on equation (3) from the main paper:

$$q(\mathbf{x}_T \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_T; \sqrt{\bar{\alpha}_T}\mathbf{x}_0, (1 - \bar{\alpha}_T)\mathbf{I}), \tag{30}$$

we have:

$$h(\mathbf{x}_T \mid \mathbf{x}_0) = \frac{1}{2}\log(2\pi e)^N(1 - \bar{\alpha}_T). \tag{31}$$

Based on equation (4) from the main paper:

$$h(\mathbf{x}_T) = \frac{1}{2}\log(2\pi e)^N. \tag{32}$$

Substituting into equation (29), we get:

$$p(e) \geq \frac{\frac{1}{2}\log(2\pi e)^N(1 - \bar{\alpha}_T) - \frac{1}{2}\log(2\pi e)^N - 1}{h(\mathbf{x}_0)} + 1 \tag{33}$$

$$= \frac{\frac{1}{2}\log(1 - \bar{\alpha}_T) - 1}{h(\mathbf{x}_0)} + 1. \tag{34}$$

Regardless of the distribution of $\mathbf{x}_0$, assume it has covariance matrix $\mathbf{K}_{\mathbf{x}_0}$. Then, its entropy has an upper bound [9]:

$$h(\mathbf{x}_0) \leq \frac{1}{2}\log(2\pi e)^N|\mathbf{K}_{\mathbf{x}_0}|, \tag{35}$$

with equality if and only if $\mathbf{x}_0$ has a Gaussian distribution. Thus, $p(e)$ is lower bounded by:

$$p(e) \geq \frac{\frac{1}{2}\log(1 - \bar{\alpha}_T) - 1}{\frac{1}{2}\log(2\pi e)^N|\mathbf{K}_{\mathbf{x}_0}|} + 1 \tag{36}$$

$$= \frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N|\mathbf{K}_{\mathbf{x}_0}|} + 1. \tag{37}$$

To make $p(e)$ equal to 0, the right-hand side of equation (37) must satisfy:

$$\frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N|\mathbf{K}_{\mathbf{x}_0}|} + 1 \leq 0. \tag{38}$$

This condition ensures that $p(e)$ is bounded away from 0, making it possible for $p(e) = 0$, which implies $\hat{\mathbf{x}}_0$ matches $\mathbf{x}_0$. □

B Proof of Corollary 1

Proof. For a DDPM, we have:

$$q(\mathbf{x}_T) \approx \mathcal{N}(\mathbf{x}_t; \mathbf{0}, \mathbf{I}), \tag{39}$$

so for $t = T$ in (3), we obtain:

$$\bar{\alpha}_T \approx 0. \tag{40}$$

Based on Theorem 1, equation (9) becomes:

$$\begin{aligned} p(e) &\geq \frac{\log(1-0)-2}{\log(2\pi e)^N|\mathbf{K}_{\mathbf{x_0}}|} + 1 && (41)\\ &= 1 - \frac{2}{\log(2\pi e)^N|\mathbf{K}_{\mathbf{x_0}}|}. && (42) \end{aligned}$$

If:

$$\frac{2}{\log(2\pi e)^N|\mathbf{K}_{\mathbf{x_0}}|} \leq 0, \tag{43}$$

then:

$$p(e) \geq 1, \tag{44}$$

which implies that $\mathbf{x}_0 \neq \hat{\mathbf{x}}_0$. To satisfy equation (43), it is required that:

$$\begin{aligned} &\log(2\pi e)^N|\mathbf{K}_{\mathbf{x_0}}| < 0, && (45)\\ &(2\pi e)^N|\mathbf{K}_{\mathbf{x_0}}| < 1, && (46)\\ &|\mathbf{K}_{\mathbf{x_0}}| < \frac{1}{(2\pi e)^N}. && (47) \end{aligned}$$

□

C Proof of Theorem 2

Proof. Based on the estimation error and differential entropy theorem in [9]:

$$\min_{\hat{\mu}_t, t=1,2,\ldots,T} \mathbb{E}_t \mathbb{E}_{q(x_0,x_t)} \left\| \boldsymbol{\epsilon}_t - \hat{\boldsymbol{\epsilon}}_t \right\|_2^2 \geq \frac{1}{(2\pi e)^N} e^{2h(\boldsymbol{\epsilon})}, \tag{48}$$

where $h(\boldsymbol{\epsilon})$ is the entropy of $\boldsymbol{\epsilon}$, and N is the dimension of the noise vector $\boldsymbol{\epsilon}$. Equality holds when $\boldsymbol{\epsilon}$ follows a Gaussian distribution, which is the case for $\boldsymbol{\epsilon}$ in DDPMs.

$$h(\boldsymbol{\epsilon}) = \frac{1}{2} \log(2\pi e)^N |\mathbf{K}_n|, \tag{49}$$

where $|\mathbf{K}_n|$ is the determinant of covariance matrix of $\boldsymbol{\epsilon}$. Observing the optimal covariance matrices for white Gaussian noise in (19) and for colored Gaussian noise in (20), they share the same diagonal values. Based on Hadamard's inequality [9]:

$$\begin{aligned} |\mathbf{K}^*_{\mathbf{n}}(\mathbf{x}_t)| &\leq \prod_{i=1}^{N} \mathbf{K}^*_{\mathbf{n}ii}(\mathbf{x}_t) && (50)\\ &= \prod_{i=1}^{N} \tilde{\boldsymbol{\sigma}}^*_n(\mathbf{x}_t)_i^2, && (51) \end{aligned}$$

colored Gaussian noise has a smaller determinant, indicating lower entropy and, therefore, a smaller MSE. □

D Proof of Corollary 2

Proof. As shown in [9]:

$$h(\boldsymbol{\epsilon}) \leq \frac{1}{2} \log(2\pi e)^N |\mathbf{K}_n|, \tag{52}$$

with equality if and only if $\boldsymbol{\epsilon}$ follows a Gaussian distribution. Therefore, non-Gaussian noise has lower entropy. Based on equation (48), a variable with lower entropy results in lower MSE, meaning that non-Gaussian noise performs better than Gaussian noise in minimizing the MSE in (17) if they have the same covariance matrix. □

E Proof of Theorem 3

Proof. For a DDPM, the variables in the forward diffusion process $\mathbf{x}_0, \ldots, \mathbf{x}_t, \ldots, \mathbf{x}_T$ and the variables in the reverse denoising process $\hat{\mathbf{x}}_T, \ldots, \hat{\mathbf{x}}_t, \ldots, \hat{\mathbf{x}}_0$ form the following Markov chain:

$$\mathbf{x}_0 \to \cdots \to \mathbf{x}_t \to \cdots \to \mathbf{x}_T \to \hat{\mathbf{x}}_T \to \cdots \hat{\mathbf{x}}_t \to \cdots \to \hat{\mathbf{x}}_0. \tag{53}$$

These variables can be represented as either matrices or vectors. If they are matrices, they can be vectorized [5], with the vector length assumed to be N.

Define the error e in $p_t(e)$ as:

$$e = \begin{cases} 1 & \text{if } \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\| \geq \epsilon, \\ 0 & \text{if } \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\| < \epsilon. \end{cases} \tag{54}$$

where ϵ is a very small number. Using Fano's inequality [9, 12], we have:

$$p_t(e) \geq \frac{h(\mathbf{x}_0 \mid \mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{55}$$

$$= \frac{h(\mathbf{x}_0, \mathbf{x}_T) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{56}$$

$$= \frac{h(\mathbf{x}_T \mid \mathbf{x}_0) + h(\mathbf{x}_0) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} \tag{57}$$

$$= \frac{h(\mathbf{x}_T \mid \mathbf{x}_0) - h(\mathbf{x}_T) - 1}{h(\mathbf{x}_0)} + 1. \tag{58}$$

Following similar steps as in the proof of Theorem 1, we can demonstrate that:

$$p_t(e) \geq \frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N |\mathbf{K}_{\mathbf{x}_0}|} + 1. \tag{59}$$

□

F Proof of Theorem 4

Proof. In a DDPM, the noise $\boldsymbol{\epsilon}_t$ can be obtained based on equation (12):

$$\boldsymbol{\epsilon}_t = \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{\bar{\beta}_t}}, \tag{60}$$

which is used to generate $\mathbf{x}_t$. The covariance matrix for noise $\boldsymbol{\epsilon}_t$ given $\mathbf{x}_t$ is computed as [3]:

$$\mathbf{K_n} = \mathrm{Cov}_{q(\mathbf{x}_0|\mathbf{x}_t)} \left(\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{\bar{\beta}_t}} \right) \tag{61}$$

$$= \frac{\bar{\alpha}_t}{\bar{\beta}_t} \mathbf{K_{x_0}}, \tag{62}$$

so

$$\mathbf{K_{x_0}} = \frac{\bar{\beta}_t}{\bar{\alpha}_t} \mathbf{K_n}. \tag{63}$$

Based on this result, the lower bound in equation (21) from Theorem 3 becomes:

$$p_t(e) \geq \frac{\log(1 - \bar{\alpha}_T) - 2}{\log(2\pi e)^N \frac{\bar{\beta}_t}{\bar{\alpha}_t} |\mathbf{K_n}|} + 1 \tag{64}$$

$$\approx 1 - \frac{2}{\log(2\pi e)^N \frac{\bar{\beta}_t}{\bar{\alpha}_t} |\mathbf{K_n}|} \tag{65}$$

$$= 1 - \frac{2\bar{\alpha}_t}{\log(2\pi e)^N \bar{\beta}_t |\mathbf{K_n}|}. \tag{66}$$

From equation (64) to (65) is based on $\bar{\alpha}_T \approx 0$. For $|\mathbf{K_n}|$ in equation (66), we have shown in equation (51) that colored Gaussian noise has a smaller determinant, so the lower bound in equation (66) is smaller for colored Gaussian noise. □

G Proof of Corollary 3

Proof. At $t = 0$, $\bar{\beta}_t \to 0$, so $\frac{2\bar{\alpha}_t}{\log(2\pi e)^N \bar{\beta}_t |\mathbf{K_n}|} \to \infty$, meaning that the right-hand side of equation (66) is less than 0. Thus, it is possible for $p_t(e) = 0$, indicating that $\mathbf{x}_0$ and $\hat{\mathbf{x}}_0$ could match each other. □

H Proof of Theorem 5

Proof. **Step 1 (Probability flow ODE).** For the VP SDE in diffusion models, the forward-time marginal q_t solves a Fokker–Planck equation. The corresponding *probability flow ODE* has drift $\mathbf{v}_t(\mathbf{x}) = \mathbf{f}_t(\mathbf{x}) - g(t)^2 \nabla_{\mathbf{x}} \log q_t(\mathbf{x})$, where for VP, $\mathbf{f}_t(\mathbf{x}) = -\frac{1}{2}\beta(t)\,\mathbf{x}$ and $g(t)^2 = \beta(t)$. The ODE $\dot{\mathbf{x}} = \mathbf{v}_t(\mathbf{x})$ induces a (deterministic) flow $\Phi_{T\to t}$ on $\mathbb{R}^d$. Let μ_t be the pushforward of the *arbitrary* initial law r_T by this flow: $\mu_t = (\Phi_{T\to t})_{\#} r_T$. By construction μ_t satisfies the continuity equation

$$\partial_t \mu_t + \nabla\cdot\big(\mu_t\, \mathbf{v}_t\big) = 0. \tag{67}$$

The data-corrupted marginal q_t (the target at noise level t) also satisfies the *same* continuity equation but with its own initial condition $q_T = \mathcal{N}(\mathbf{0}, \mathbf{I})$ (for standard VP schedules).

Step 2 (KL dissipation along the flow). Define the KL divergence $\mathcal{K}(t) = \mathrm{KL}(\mu_t \,\|\, q_t) = \int \mu_t \log\big(\mu_t/q_t\big)\, d\mathbf{x}$. Differentiate $\mathcal{K}$ with respect to t and use the product rule:

$$\frac{d}{dt}\mathcal{K}(t) = \int \partial_t \mu_t\, \log\Big(\frac{\mu_t}{q_t}\Big)\, d\mathbf{x} \;+\; \int \mu_t\, \partial_t \log\Big(\frac{\mu_t}{q_t}\Big)\, d\mathbf{x}. \tag{68}$$

Since $\partial_t \log(\mu_t/q_t) = (\partial_t\mu_t)/\mu_t - \partial_t \log q_t$, the second integral becomes $\int \partial_t\mu_t\, d\mathbf{x} - \int \mu_t\, \partial_t \log q_t\, d\mathbf{x} = -\int \mu_t\, \partial_t \log q_t\, d\mathbf{x}$ because $\int \partial_t\mu_t = \partial_t \int \mu_t = 0$. Insert (67) and integrate by parts in the first term of (68):

$$\frac{d}{dt}\mathcal{K}(t) = \int \big[-\nabla\cdot(\mu_t \mathbf{v}_t)\big]\, \log\Big(\frac{\mu_t}{q_t}\Big)\, d\mathbf{x} \;-\; \int \mu_t\, \partial_t \log q_t\, d\mathbf{x} \tag{69}$$

$$= \int \mu_t\, \mathbf{v}_t \cdot \nabla \log\Big(\frac{\mu_t}{q_t}\Big)\, d\mathbf{x} \;-\; \int \mu_t\, \partial_t \log q_t\, d\mathbf{x}. \tag{70}$$

Using $\mathbf{v}_t = \mathbf{f}_t - g(t)^2\, \nabla \log q_t$,

$$\int \mu_t\, \mathbf{v}_t \cdot \nabla \log\Big(\frac{\mu_t}{q_t}\Big) = \int \mu_t\, \mathbf{f}_t \cdot \big(\nabla \log \mu_t - \nabla \log q_t\big)\, d\mathbf{x} \;-\; g(t)^2 \int \mu_t\, \|\nabla \log \mu_t - \nabla \log q_t\|^2\, d\mathbf{x}.$$

Next, recall that q_t satisfies the same continuity equation $\partial_t q_t + \nabla\cdot(q_t \mathbf{v}_t) = 0$. Dividing by q_t gives $\partial_t \log q_t = -\nabla\cdot \mathbf{v}_t - \mathbf{v}_t \cdot \nabla \log q_t$. Multiplying by μ_t and integrating yields

$$-\int \mu_t\, \partial_t \log q_t\, d\mathbf{x} = \int \mu_t\, \nabla\cdot \mathbf{v}_t\, d\mathbf{x} + \int \mu_t\, \mathbf{v}_t \cdot \nabla \log q_t\, d\mathbf{x}.$$

Combine these identities in (70) and use that $\nabla\cdot(\mu_t \mathbf{f}_t) = \mu_t\, \nabla\cdot \mathbf{f}_t + \mu_t\, \mathbf{f}_t \cdot \nabla \log \mu_t$ to cancel all terms involving $\mathbf{f}_t$ and divergences (standard calculus under vanishing boundary conditions). One obtains the *dissipation identity*

$$\frac{d}{dt}\mathcal{K}(t) = -\,g(t)^2 \underbrace{\int \mu_t\, \big\|\nabla \log \mu_t - \nabla \log q_t\big\|^2\, d\mathbf{x}}_{=\ \mathcal{J}(\mu_t \,\|\, q_t)\ \geq 0}, \tag{71}$$

where $\mathcal{J}(\mu_t \,\|\, q_t)$ is the (generalized) Fisher/Stein divergence.

Step 3 (Monotonicity and endpoint). Because $\beta(t) = g(t)^2 > 0$ for $t \in (0, T]$, (71) implies $\frac{d}{dt}\mathcal{K}(t) \leq 0$ with equality iff $\nabla \log \mu_t = \nabla \log q_t$, i.e. $\mu_t = q_t$. Hence, $\mathcal{K}(t)$ is nonincreasing along the reverse-time integration and $\lim_{t\downarrow 0} \mathcal{K}(t) = \mathcal{K}(0) = 0$ because q_0 is the data distribution and the flow is well-posed with an exact score.

Step 4 (Independence from the terminal law). The derivation above makes no assumption on the *shape* of the initial law r_T beyond mild regularity (density, finite second moment) to justify integrations by parts. In particular, r_T may be *any* symmetric zero-mean distribution (not necessarily Gaussian). The probability flow dynamics uses only the *pointwise* score $\nabla \log q_t$ and schedule $g(t)$; the initial law affects only which trajectories are instantiated, not the form of the denoising vector field. Therefore the sampler *effectively denoises* from any such r_T: the discrepancy to the target marginal strictly decreases according to (71) and vanishes at $t = 0$.

This proves the claim. □

Remarks. (1) The proof uses the deterministic probability-flow sampler (DDIM, $\eta = 0$). With stochastic DDPM sampling ($\eta > 0$), an analogous KL decay can be shown at the level of the reverse SDE using standard score-based diffusion arguments. (2) The *zero-mean symmetry* of r_T is natural in practice (it preserves centering) but not essential for the KL dissipation identity (71); it ensures well-behaved moments and unbiased initialization.

I Additional Experiments

We evaluate the performance of the DDPM trained with zero-mean, unit-variance white Gaussian noise by testing its output against six types of non-Gaussian noise. To ensure a fair comparison, we designed 2D noise with an average power of 2 (with a power of 1 in each dimension).

Below, we provide a brief introduction to four symmetric non-Gaussian noises:

1. The probability density function (PDF) of the Uniform Distribution in each dimension is given by [19]:
$$f(x) = \begin{cases} \frac{1}{2\sqrt{3}}, & x \in [-\sqrt{3}, \sqrt{3}], \\ 0, & \text{otherwise.} \end{cases} \tag{72}$$
This distribution has a power of 1 in each dimension.
2. The PDF of the Laplace Distribution is expressed as [19]:
$$f(x \mid \mu, b) = \frac{1}{2b} \exp\left(-\frac{|x-\mu|}{b}\right), \tag{73}$$
where μ is the location parameter (mean), and b is the scale parameter. We set $\mu = 0$ and $b = \frac{1}{\sqrt{2}}$ to ensure that the average power is 1 in each dimension.
3. The PDF of the Cosine Distribution is given by [19]:
$$f(x) = \begin{cases} \frac{1}{2b}\left[1 + \cos\left(\frac{\pi(x-a)}{b}\right)\right], & a - b \le x \le a + b, \\ 0, & \text{otherwise.} \end{cases} \tag{74}$$
Here, a is the location parameter (mean), and b is the scale parameter (controlling the spread). This distribution is symmetric around a and is confined to the interval $[a - b, a + b]$. The Cosine Distribution's compact support and smoothness make it useful in applications involving directional data and circular statistics.
4. The PDF of the Logistic Distribution is defined as [19]:
$$f(x; \mu, s) = \frac{e^{-\frac{x-\mu}{s}}}{s\left(1 + e^{-\frac{x-\mu}{s}}\right)^2}, \tag{75}$$
where μ is the location parameter (analogous to the mean), and s is the scale parameter (controlling the spread). The Logistic Distribution is symmetric around its mean μ, similar to the normal distribution, but with heavier tails. This property makes it useful in scenarios where outliers are expected, or where a distribution with fatter tails than the normal distribution is needed.

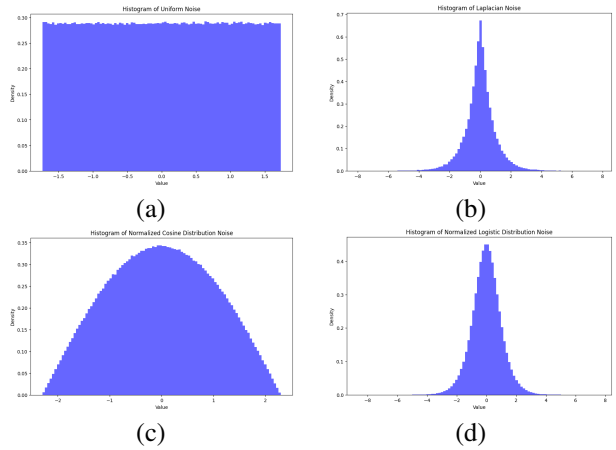


Figure 5: Histograms of symmetric non-Gaussian noises used for DDPM evaluation. (a) Uniform distribution; (b) Laplace distribution; (c) Cosine distribution; (d) Logistic distribution.

In Fig. 5, we present the histograms of these four symmetric non-Gaussian noises, all of which have zero-mean and unit power in each dimension.

We also introduce two asymmetric non-Gaussian noises:

1. The PDF of the Gamma Distribution is given by [19]:

$$f(x;\alpha,\beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \quad \text{for } x > 0, \tag{76}$$

   where $\alpha > 0$ is the shape parameter, $\beta > 0$ is the rate parameter, and $\Gamma(\alpha)$ is the Gamma function. The Gamma Distribution is versatile and can take various shapes depending on the values of α and β. It is widely used to model phenomena where the occurrence rate varies over time.

2. The PDF of the Exponential Distribution is given by [19]:

$$f(x;\lambda) = \lambda e^{-\lambda x}, \quad \text{for } x \geq 0, \tag{77}$$

   where $\lambda > 0$ is the rate parameter, which is the inverse of the mean. The Exponential Distribution is memoryless, meaning the probability of an event occurring in the next interval is independent of how much time has already passed.

In Fig. 6, we present the histograms of these two asymmetric non-Gaussian noises, both with zero-mean and unit power in each dimension.

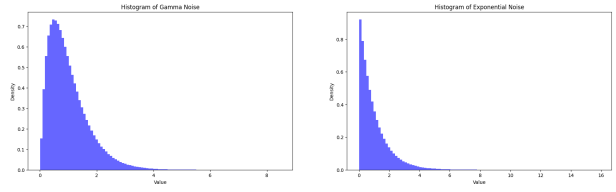


Figure 6: Histograms of asymmetric non-Gaussian noises used for DDPM evaluation. (a) Gamma distribution; (b) Exponential distribution.

In Fig. 7, we present the outputs generated by the DDPM using six different non-Gaussian noise distributions.

As shown in Fig. 7, the zero-mean symmetric noise distributions successfully generated high-resolution images, whereas the asymmetric noise distributions failed to produce any images. These

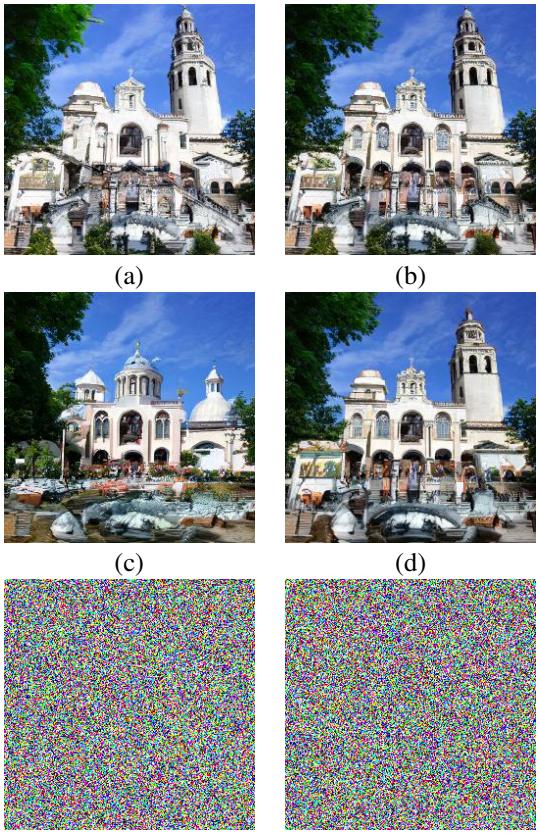


Figure 7: Generated outputs of DDPM with zero-mean non-Gaussian noises. (a) Uniform distribution noise, (b) Laplacian distribution noise, (c) Cosine distribution noise, (d) Logistic distribution noise, (e) Gamma distribution noise, (f) Exponential distribution noise.

experiments demonstrate that noise samples with a zero-mean symmetric distribution can generate high-quality images, even when the DDPM was trained with white Gaussian noise, thus confirming Theorem 5.

The above experiments on LSUN Church-256 were performed using Google Colab with T4 GPU. The total running time was around 60 minutes.