

Golden Noise for Diffusion Models: A Learning Framework

Zikai Zhou¹, Shitong Shao¹, Lichen Bai¹, Shufei Zhang², Zhiqiang Xu³, Bo Han⁴, Zeke Xie^{1*}
¹ HKUST-GZ ² Shanghai AI Lab ³ MBZUAI ⁴ HKBU

{zikaizhou, sshao213, lichenbai, zekexie}@hkust-gz.edu.cn

zhangshufei@pjlab.org.cn

zhiqiang.xu@mbzuai.ac.ae

bhanml@comp.hkbu.edu.hk

Github: <https://github.com/xie-lab-ml/Golden-Noise-for-Diffusion-Models>

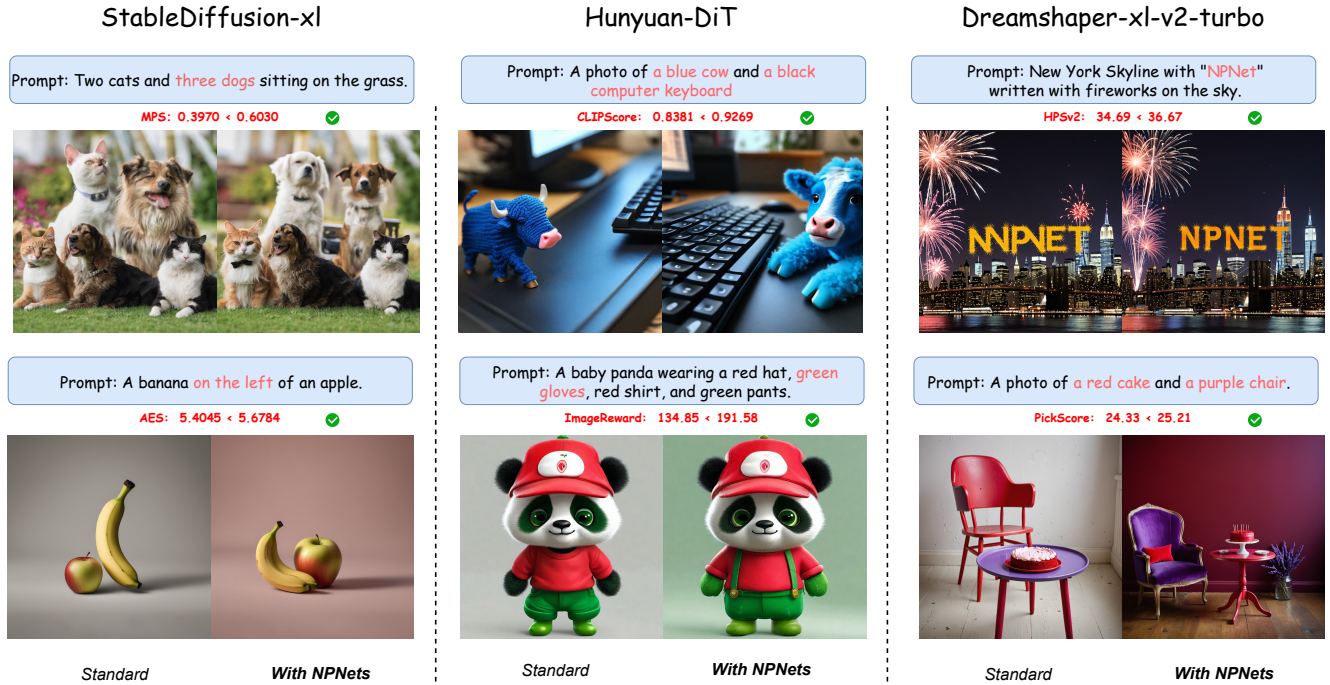


Figure 1. We visualize images synthesized by 3 different diffusion models and evaluate them using 6 human preference metrics. Images for each prompt are synthesized using the same random seed. These images with NPNet demonstrate a noticeable improvement in overall quality, aesthetic style, and semantic faithfulness, along with numerical improvements across all six metrics. More importantly, our NPNet is applicable to various diffusion models, showcasing strong generalization performance with broad application potential. More visualization results are in Appendix 22.

Abstract

Text-to-image diffusion model is a popular paradigm that synthesizes personalized images by providing a text prompt and a random Gaussian noise. While people observe that some noises are “golden noises” that can achieve better text-image alignment and higher human preference than others, we still lack a machine learning framework to obtain those golden noises. To learn golden noises for diffusion sampling, we mainly make three contributions in this paper. First, we identify a new concept termed the noise prompt, which aims at turning a random Gaussian noise

into a golden noise by adding a small desirable perturbation derived from the text prompt. Following the concept, we first formulate the noise prompt learning framework that systematically learns “prompted” golden noise associated with a text prompt for diffusion models. Second, we design a noise prompt data collection pipeline and collect a large-scale noise prompt dataset (NPD) that contains 100k pairs of random noises and golden noises with the associated text prompts. With the prepared NPD as the training dataset, we trained a small noise prompt network (NPNet) that can directly learn to transform a random noise into a golden noise. The learned golden noise perturbation can be con-

sidered as a kind of prompt for noise, as it is rich in semantic information and tailored to the given text prompt. Third, our extensive experiments demonstrate the impressive effectiveness and generalization of NPNet on improving the quality of synthesized images across various diffusion models, including SDXL, DreamShaper-xl-v2-turbo, and Hunyuan-DiT. Moreover, NPNet is a small and efficient controller that acts as a plug-and-play module with very limited additional inference and computational costs, as it just provides a golden noise instead of a random noise without accessing the original pipeline.

1. Introduction

Image synthesis that are precisely aligned with given text prompts remains a significant challenge for text-to-image (T2I) diffusion models [4, 7, 35, 36, 42]. Previous studies [24, 28, 50, 52, 57] have investigated the influence of text embeddings on the synthesized images and leveraged these embeddings for training-free image synthesis. It is well known that text prompts significantly matter to the quality and fidelity of the synthesized images. However, image synthesis is induced by both the text prompts and the noise. Variations in the noise can lead to substantial changes in the synthesized images, as even minor alterations in the noise input can dramatically influence the output [38, 56]. This sensitivity underscores the critical role that noise plays in shaping the final visual representation, affecting both the overall aesthetics and the semantic faithfulness between the synthesized images and the provided text prompt.

Recent studies [2, 5, 8, 11, 32, 38] observe that some selected or optimized noises are golden noises that can help the T2I diffusion models to produce images of better semantic faithfulness with text prompts, and can also improve the overall quality of the synthesized images. These methods [5, 11] incorporate extra modules like attention to reduce the truncate errors during the sampling process, showing promising results on the compositional generalization task. However, they are often not widely adopted in practice for several reasons. First, they often struggle to generally transfer to various benchmark datasets or diffusion models but only work for some specific tasks. Second, these methods often introduce significant time delays in order to optimize the noises during the reverse process. Third, they require in-depth modifications to the original pipelines when applied to different T2I diffusion models with varying architectures. Fourth, they need specific subject tokens for each prompt to calculate the loss of certain areas, which are unrealistic requirements for real users. These not only significantly complicate the original inference pipeline but also raise concerns regarding the generalization ability across various T2I diffusion models and datasets.

In light of the aforementioned research, we pose sev-

eral critical questions: 1) Can we formulate obtaining the golden noises as a machine learning problem so that we can predict them efficiently with only one model forward inference? 2) Can such a machine learning framework generalize well to various noises, prompts, and even diffusion models? Fortunately, the answers are affirmative. In this paper, we mainly make three contributions:

First, we identify a new concept termed *noise prompt*, which aims at turning a random noise into a golden noise by adding a small desirable perturbation derived from the text prompt. The golden noise perturbation can be considered as a kind of prompt for noise, as it is rich in semantic information and tailored to the given text prompt. Building upon this concept, we formulate a *noise prompt learning* framework that learns “prompted” golden noises associated with text prompts for diffusion models.

Second, to implement the formulated *noise prompt learning* framework, we propose the training dataset, namely the *noise prompt dataset* (NPD), and the learning model, namely the *noise prompt network* (NPNet). Specifically, we design a noise prompt data collection pipeline via *re-denoise sampling*, a way to produce noise pairs. We also incorporate AI-driven feedback mechanisms to ensure that the noise pairs are highly valuable. This pipeline enables us to collect a large-scale training dataset for noise prompt learning, so the trained NPNet can directly transform a random Gaussian noise into a golden noise to boost the performance of the T2I diffusion model.

Third, we conduct extensive experiments across various mainstream diffusion models, including StableDiffusion-xl (SDXL) [37], DreamShaper-xl-v2-turbo and Hunyuan-DiT [25], with 7 different samplers on 4 different datasets. We evaluate our model by utilizing 6 human preference metrics including Human Preference Score v2 (HPSv2) [53], PickScore [23], Aesthetic Score (AES) [46], ImageReward [55], CLIPScore [12] and Multi-dimensional Preference Score (MPS) [59]. As illustrated in Fig. 1, by leveraging the learned golden noises, not only is the overall quality and aesthetic style of the synthesized images visually enhanced, but all metrics also show significant improvements, demonstrating the effectiveness and generalization ability of our NPNet. Furthermore, the NPNet is a compact and efficient neural network that functions as a plug-and-play module, introducing only a **3%** extra inference time per image compared to the standard pipeline, while requiring approximately 3% of the memory required by the standard pipeline. This underscores the practical applicability of NPNet in real-world scenarios.

2. Preliminaries

We first present preliminaries about DDIM and DDIM Inversion and the classifier-free guidance. Due to the space constraints, we introduce the related work in Appendix B.

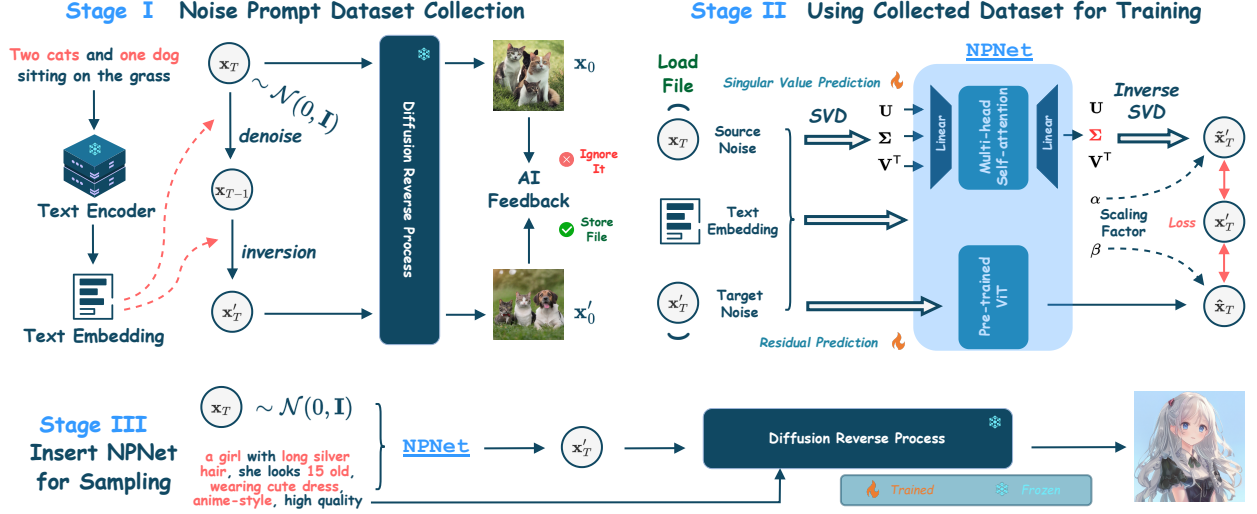


Figure 2. Our workflow diagram consists of three main stages. *Stage I*: We begin by denoising the original random Gaussian noise $\mathbf{x}_T$ to obtain $\mathbf{x}_{T-1}$, and then use DDIM-Inversion($\cdot$) to obtain inverse $\mathbf{x}'_T$ with more semantic information. Both synthesized images $\mathbf{x}_0$ and $\mathbf{x}'_0$ are filtered by the human preference model, such as HPSv2, to ensure the dataset is both diverse and representative. *Stage II*: After collecting NPD, we input the original noise (source noise) $\mathbf{x}_T$, inverse noise (target noise) $\mathbf{x}'_T$ and text prompt $\mathbf{c}$ into the NPNNet, where the noises are processed by the *singular value predictor* and the *residual predictor*, and text prompt $\mathbf{c}$ is encoded by the text encoder $\mathcal{E}(\cdot)$ of the T2I diffusion model, resulting in the golden noise. *Stage III*: Once trained, our NPNNet can directly convert the random Gaussian noise into a golden noise before inputting T2I diffusion models, boosting the performance of these models.

Given the Gaussian noise $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$, we denote the forward process of diffusion models as $\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \epsilon_t$, where the $t \in \{0, 1, \dots, T\}$. Here, α_t and σ_t are predefined noise schedules, and $\mathbf{x}_0$ is the original image.

DDIM and DDIM Inversion. Denoising diffusion implicit model (DDIM) [48] is an advanced deterministic sampling technique, deriving an implicit non-Markov sampling process of the diffusion model. It allows for faster synthesis while maintaining the quality of synthesized samples. Its reverse process can be formulated as:

$$\begin{aligned} \mathbf{x}_{t-1} &= \text{DDIM}(\mathbf{x}_t) \\ &= \alpha_{t-1} \left(\frac{\mathbf{x}_t - \sigma_t \epsilon_\theta(\mathbf{x}_t, t)}{\alpha_t} \right) + \sigma_{t-1} \epsilon_\theta(\mathbf{x}_t, t) \end{aligned} \quad (1)$$

Using DDIM to add noise is called DDIM-Inversion:

$$\begin{aligned} \mathbf{x}_t &= \text{DDIM-Inversion}(\mathbf{x}_{t-1}) \\ &= \frac{\alpha_t}{\alpha_{t-1}} \mathbf{x}_{t-1} + \left(\sigma_t - \frac{\alpha_t}{\alpha_{t-1}} \sigma_{t-1} \right) \epsilon_\theta(\mathbf{x}_t, t) \end{aligned} \quad (2)$$

Classifier-free Guidance (CFG). Classifier-free guidance [14] allows for better control over the synthesis process by guiding the diffusion model towards desired conditions, such as text prompt, to enhance the quality and diversity of synthesized samples. The predicted noise ϵ_{pred} with CFG at timestep t can be formulated as:

$$\epsilon_{pred} = (\omega + 1) \epsilon_\theta(\mathbf{x}_t, t | \mathbf{c}) - \omega \epsilon_\theta(\mathbf{x}_t, t | \emptyset), \quad (3)$$

where we denote the $\mathbf{c}$ as the text prompt, ω as the CFG scale. Based on this, the denoised image $\mathbf{x}_{t-1}$ by using DDIM($\cdot$) can be written as:

$$\begin{aligned} \mathbf{x}_{t-1} &= \alpha_{t-1} \left(\frac{\mathbf{x}_t - \sigma_t [(\omega + 1) \epsilon_\theta(\mathbf{x}_t, t | \mathbf{c}) - \omega \epsilon_\theta(\mathbf{x}_t, t | \emptyset)]}{\alpha_t} \right) \\ &\quad + \sigma_{t-1} [(\omega + 1) \epsilon_\theta(\mathbf{x}_t, t | \mathbf{c}) - \omega \epsilon_\theta(\mathbf{x}_t, t | \emptyset)] \end{aligned} \quad (4)$$

3. Noise Prompt Learning

In this section, we present the methodology of noise prompt learning, including NPD collection, NPNNet design and training, as well as sampling with NPNNet.

Noise prompt can be considered as a kind of special prompt, which aims at turning a random noise into a golden noise by adding a small desirable perturbation derived from the text prompt. Analogous to text prompts, appropriate noise prompts can enable diffusion models to synthesize higher-quality images that are rich in semantic information. As illustrated on the left in Appendix Fig. 6, *text prompt learning* in large language models [30] focuses on learning how to transform a text prompt into a more desirable version. Similarly, *noise prompt learning* in our work seeks to learn how to convert the random Gaussian noise into the golden noise by adding a small, desirable perturbation derived from the text prompt. Using the golden noise, the diffusion model can synthesize images with higher quality and semantic faithfulness. Defining it as a machine learning problem, we are the first to formulate the *noise prompt learning* framework, as illustrated on the right in Appendix

Fig. 6. Given the training set $\mathcal{D} := \{\mathbf{x}_{T_i}, \mathbf{x}'_{T_i}, \mathbf{c}_i\}_{i=1}^{|\mathcal{D}|}$ consisting of source noises $\mathcal{X}$, target noises $\mathcal{X}'$ and text prompts $\mathcal{C}$, loss function ℓ and the neural network ϕ , the general formula for the *noise prompt learning* task is:

$$\phi^* = \arg \min_{\phi} \mathbb{E}_{(\mathbf{x}_{T_i}, \mathbf{x}'_{T_i}, \mathbf{c}_i) \sim \mathcal{D}} [\ell(\phi(\mathbf{x}_{T_i}, \mathbf{c}_i), \mathbf{x}'_{T_i})]. \quad (5)$$

In summary, our goal is to learn the optimal neural network model ϕ^* trained on the training set $\mathcal{D}$. We present the data-training-inference workflow diagram with three stages in Appendix Fig. 6 and provide the pseudocodes for each stage in Appendix Alg. 1, Alg. 2 and Alg. 3, respectively.

3.1. Noise Prompt Dataset Collection

In this subsection, we outline the training data collection pipeline, which consists of collecting noise pairs and AI-feedback-based data selection, as shown in Fig. 2 Stage I.

Re-denoise Sampling Produces Noise Pairs. *How to collect noises with desirable semantic information?* Meng et al. [34] reported that adding the random noise at each timestep during the sampling process and then re-denoising, leads to a substantial improvement in the semantic faithfulness of the synthesized images. This motivated us to propose a simple and direct approach called RE-DENOISE SAMPLING. Instead of directly adding noise to each timestep during the reverse process, we propose to utilize DDIM-Inversion($\cdot$) to obtain the noise from the previous step. Specifically, the joint action of DDIM-Inversion and CFG can induce the initial noise to attach semantic information. We denote the CFG scale within DDIM($\cdot$) and DDIM-Inversion($\cdot$) as ω_l and ω_w , respectively. It is sufficient to ensure that the initial noise can be purified stably and efficiently by $\mathbf{x}'_t = \text{DDIM-Inversion}(\text{DDIM}(\mathbf{x}_t))$ with $\omega_l > \omega_w$. Utilizing this method, the synthesized image from $\mathbf{x}'_t$ is more semantic information contained with higher fidelity, compared with the synthesized image from $\mathbf{x}_t$. We call the inverse noise $\mathbf{x}'_t$ target noise, and the noise $\mathbf{x}_t$ source noise. The visualization results are shown in Appendix Fig. 7 and 9. The mechanism behind this method is that DDIM-Inversion($\cdot$) injects semantic information by leveraging the CFG scale inconsistency. We present **theoretical understanding** of this mechanism in Theorem E.1.

Data Selection with the Human Preference Model.

While employing *re-denoise sampling* can help us collect noises with enhanced semantic information, it also carries the risk of introducing extra noises, which may lead to synthesizing images that do not achieve the quality of the originals. To mitigate this issue, we utilize a human preference model for data selection. This model assesses the synthesized images based on human preferences, allowing us to retain those noise samples that meet our quality standards.

The reservation probability for data selection can be formulated as $\mathbb{I}[s_0 + m < s'_0]$, where m is the filtering threshold, $\mathbb{I}[\cdot]$, s_0 and s'_0 are the indicator function, human preference scores of denoised images from $\mathbf{x}_0$ and $\mathbf{x}'_0$, respectively. If the noise samples meet this criterion, we consider them to be valuable noise pairs and proceed to collect them.

By implementing this filtering process, we aim to balance the benefits of *re-denoise sampling* with the integrity of the synthesized outputs. For the selection strategies, we introduce them in Appendix Sec. D.2.

3.2. Noise Prompt Network

Here, we introduce the architecture, training, inference of NPNet, as shown in Fig. 2 Stage II and Stage III.

Architecture Design. The architecture of NPNet consists of two key components, including *singular value prediction* and *residual prediction*, as shown in Fig. 2 Stage II.

The first key model component is *singular value prediction*. We obtain noise pairs through *re-denoise sampling*, a process that can be approximated as adding a small perturbation to the source noises. We observe that through the *singular value decomposition* (SVD), the singular vectors of $\mathbf{x}_T$ and $\mathbf{x}'_T$ exhibit remarkable similarity, albeit possibly in opposite directions, shown in Appendix Fig. 8, which may be partly explained by Davis-Kahan Theorem [49, 54]. Building upon this observation, we design an architecture to predict the singular values of the target noise, illustrated in Fig. 2 Stage II. We denote $\phi(\cdot, \cdot, \cdot)$ as a ternary function that represents the sum of three inputs, $f(\cdot)$ as the linear layer function, and $g(\cdot)$ as the multi-head self-attention layer. The paradigm can be formulated as:

$$\begin{aligned} \mathbf{x}_T &= U \times \Sigma \times V^T, & \tilde{\mathbf{x}}_T &= \phi(U, \Sigma, V^T), \\ \tilde{\Sigma} &= f(g(\tilde{\mathbf{x}}_T)), & \tilde{\mathbf{x}}'_T &= U \times \tilde{\Sigma} \times V^T, \end{aligned} \quad (6)$$

where we denote $\tilde{\mathbf{x}}'_T$ as the predicted target noise. This model utilizes SVD inverse transformation to effectively reconstruct the target noise. By leveraging the similarities in the singular vectors, our model enhances the precision of the target noise restoration process.

The second key model component is residual prediction. In addition to *singular value prediction*, we also design an architecture to predict the residual between the source noise and the target noise, as illustrated in Fig. 2 Stage II. We denote $\varphi(\cdot)$ as the UpSample-DownConv operation, $\varphi'(\cdot)$ as the DownSample-UpConv operation, and the $\psi(\cdot)$ as the ViT model. The target noise incorporates semantic information from the text prompt $\mathbf{c}$ introduced through CFG. To facilitate the learning process, we inject this semantic information using the frozen text encoder $\mathcal{E}(\cdot)$ of the T2I diffusion model. This approach allows the model to effectively leverage the semantic information provided by the

Table 1. Summary of the experiments in this paper.

Experiments		Results
Main Experiments		Table 2, 3, Appendix Table 8 and 13 Fig. 3, Appendix Fig. 12, 14 and 17
Generalization	Various Datasets and Models	Table 2, 3, Appendix Table 6, 19 and 18 Fig. 3
	Stochastic/Deterministic Samplers	Appendix Table 11 Fig. 5 and Appendix Fig. 10
	Orthogonal Experiment	Table 5, Appendix Table 20 and Fig. 19
Robustness to Hyper-parameters		Appendix Table 12 and 17
		Fig. 4, Appendix Fig. 14, 16 and 17
Ablation Studies		Table 4, Appendix Table 9 and 10
Efficiency		Appendix Table 11
		Fig. 5 and Appendix Fig. 10
Downstream Task		Appendix Fig. 20
Visualization		Fig. 1, Appendix Fig. 20, 22, 24, 25, 26, 27, 28, 29, 31, 30, 32 and 33.

text prompt, ultimately improving the accuracy of the residual prediction. The procedure can be described as follows:

$$\mathbf{e} = \sigma(\mathbf{x}_T, \mathcal{E}(\mathbf{c})) \quad \hat{\mathbf{x}}_T = \varphi'(\psi(\varphi(\mathbf{x}_T + \mathbf{e}))), \quad (7)$$

where we denote $\sigma(\cdot, \cdot)$ as AdaGroupNorm to stabilize the training process, and $\mathbf{e}$ as the normalized text embedding.

Using Collected Dataset for Training. The training procedure is also illustrated in Fig. 2 *Stage II*. To yield optimal results, we formulate the training loss as

$$\begin{aligned} \mathcal{L}_{\text{MSE}} &= \text{MSE}(\mathbf{x}'_T, \mathbf{x}'_{T_{\text{pred}}}), \\ \text{where } \mathbf{x}'_{T_{\text{pred}}} &= \tilde{\mathbf{x}}'_T + \beta \hat{\mathbf{x}}_T, \end{aligned} \quad (8)$$

β is a trainable parameter used to adaptively adjust the weights of the predicted residuals, and L as the $\text{MSE}(\cdot)$ loss function. For the nuanced adjustment in how much semantic information contributes to the model’s predictions, $\mathbf{x}'_{T_{\text{pred}}} = \tilde{\mathbf{x}}'_T + \beta \hat{\mathbf{x}}_T$ can be rewritten as:

$$\mathbf{x}'_{T_{\text{pred}}} = \alpha \mathbf{e} + \tilde{\mathbf{x}}'_T + \beta \hat{\mathbf{x}}_T, \quad (9)$$

where we denote α as a trainable parameter. The values of these two parameters are shown in Appendix Table 15. we notice that α is very small, but it still plays a role in adjusting the influence of injected semantic information (see Appendix Table 16 for experimental results). Together, α and β finely control the impact of semantic information on the model’s predictions, enhancing both the semantic relevance between the text prompt and synthesized images and the diversity of these images.

Insert NPNet for Sampling. The inference procedure is illustrated in Fig. 2 *Stage III*. Once trained, our NPNet can be directly applied to the T2I diffusion model by inputting the initial noise $\mathbf{x}_T$ and prompt embedding $\mathbf{c}$ encoded by

the frozen text encoder of the diffusion model. Our NPNet can effectively transform the original initial noise into the golden noise. We also provide the example code on SDXL, shown in the Appendix Fig. 21.

4. Empirical Analysis

In this section, we empirically study the effectiveness, the generalization, and the efficiency of our NPNet. We conduct a lot of experiments across various datasets on various T2I diffusion models, including SDXL, DreamShaper-xl-v2-turbo, and Hunyuan-DiT. Due to space constraints, we leave implementation details, related work and additional experiments in Appendix A, C and D, respectively. We also leave the discussion and future direction in Appendix F. To assist readers in better understanding this paper, we have summarized the main tables and figures in Table 1.

Description of Training and Test Data. We collect our NPD on Pick-a-pic dataset [23], which contains 1M prompts in its training set. We randomly choose 100k prompts as our training set. For each prompt, we randomly assign a seed value in $[0, 1024]$. For testing, we use three datasets, including the first 100 prompts from the Pick-a-Pic web application, the first 100 prompts from HPD v2 test set [53], all 200 prompts from DrawBench [45], all prompts from GenEval [10] and T2I-CompBench [20]. For more details about the test data, please see Appendix Fig. 18. We construct three training datasets collected from Pick-a-Pic, with 100k noise pairs, 80k noise pairs, and 600 noise pairs for SDXL, DreamShaper-xl-v2-turbo, and Hunyuan-DiT.

4.1. Main Results

We evaluate our NPNet in three different T2I diffusion models. The main results¹ are shown in Table 2 and Table 3. In Table 2, we first evaluate our NPNet with SDXL and DreamShaper-xl-v2-turbo. The results demonstrate the impressive performance of our NPNet, almost achieving the

¹Method “Inversion” means *re-denoise sampling*.

Table 2. Main experiments on SDXL and DreamShaper-xl-v2-turbo over various datasets. Note that MPS calculates the preference scores between two images. We choose the standard sampling as the baseline and the MPS score is always 50%, marked as “—”.

Model	Dataset	Method	PickScore (↑)	HPSv2 (↑)	AES (↑)	ImageReward (↑)	CLIPScore (↑)	MPS(%) (↑)
SDXL	Pick-a-Pic	Standard	21.69	28.48	6.0373	58.01	0.8204	-
		Inversion ¹	21.71	28.57	6.0503	63.27	0.8250	51.41
		NPNNet (ours)	21.86	28.68	6.0540	65.01	0.8408	52.14
	DrawBench	Standard	22.31	26.72	5.5952	62.21	0.8077	-
		Inversion	22.37	26.91	5.6017	67.09	0.8081	51.98
		NPNNet (ours)	22.38	27.14	5.6034	70.67	0.8153	53.70
	HPD	Standard	22.88	29.71	5.9985	96.63	0.8734	-
		Inversion	22.89	29.78	5.9948	97.39	0.8708	53.03
		NPNNet (ours)	22.94	29.88	5.9922	98.81	0.8813	56.02
	DreamShaper-xl-v2-turbo	Standard	22.41	32.12	6.0161	98.09	0.8267	-
DreamShaper-xl-v2-turbo	Pick-a-Pic	Inversion	22.40	32.03	6.0236	100.97	0.8277	49.14
		NPNNet (ours)	22.73	32.69	6.0646	106.74	0.8958	52.34
	DrawBench	Standard	22.98	30.39	5.6735	98.84	0.8186	-
		Inversion	22.94	30.10	5.6852	96.74	0.8189	46.62
		NPNNet (ours)	23.11	30.78	5.7005	108.14	0.8224	53.53
	HPD	Standard	23.68	30.96	6.1408	129.89	0.8868	-
		Inversion	23.67	31.00	6.0811	131.80	0.8912	46.94
		NPNNet (ours)	23.70	34.08	6.1283	135.98	0.8942	52.49

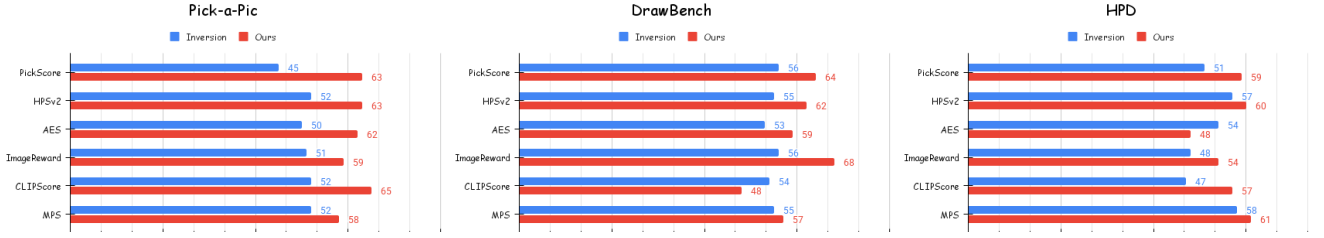


Figure 3. The winning rate comparison on SDXL across 3 datasets, including Pick-a-Pic, DrawBench and HPD v2 (HPD). The results reveal that our NPNNet is the only one that can consistently transform random Gaussian noise into golden noises, thereby enhancing the quality of the synthesized images, across nearly all evaluated datasets and metrics.

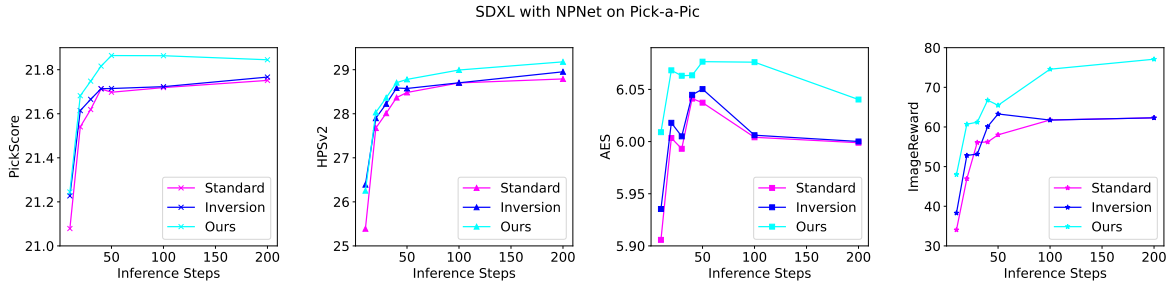


Figure 4. Visualization of performance w.r.t. inference steps on SDXL on Pick-a-Pic dataset. With our NPNNet, T2I diffusion models can have superior performance under various inference steps.

Table 3. The fine-tuned NPNNet showing strong cross-model generalization to enhancing Hunyuan-DiT, requiring only Hunyuan-DiT-produced 600 noise pairs for fine-tuning.

Dataset	Method	PickScore (↑)	HPSv2 (↑)	AES (↑)	ImageReward (↑)	CLIPScore(%) (↑)	MPS(%) (↑)
Pick-a-Pic	Standard	21.82	29.82	6.2850	91.33	80.37	-
	Inversion	21.76	29.64	6.2756	88.77	80.21	49.08
	NPNNet (ours)	21.84	29.94	6.3470	100.82	81.01	51.60
DrawBench	Standard	22.44	28.75	5.7152	91.30	79.40	-
	Inversion	22.44	28.75	5.7522	92.86	79.55	49.25
	NPNNet (ours)	22.45	28.89	5.8234	96.20	80.75	51.93
HPD	Standard	22.89	30.87	6.0793	99.22	85.68	-
	Inversion	22.90	30.56	6.0802	100.21	86.17	51.63
	NPNNet (ours)	22.89	31.20	6.1573	108.29	86.94	52.87

best results across all 6 metrics and 3 datasets. Due to the space limitations, we present the experiments on GenEval and T2I-CompBench in Appendix D.7 and 8. We also evaluate our fine-tuned NPNet with Hunyuan-DiT, as shown in Table 3. For Hunyuan-DiT, we directly utilized the NPNet model trained on SDXL-produced NPD and fine-tune it on the Hunyuan-DiT-produced 600 noise pairs. Fine-tuned with only 600 samples, it can still achieve the highest results over the baselines on Hunyuan-DiT. This highlights the strong cross-model generalizability of our NPNet.

We also show the winning rate (the ratio of winning noise in the test set) on SDXL with our NPNet, shown in Fig. 3. We present more winning rate experiments of DreamShaper-xl-v2-turbo and Hunyuan-DiT in Appendix Fig. 14 and Appendix Fig. 17. These results again support that our NPNet is highly effective in transforming random Gaussian noise into golden noises. In order to validate the effectiveness of our NPNet on improving the conventional image quality metric, we also calculate the FID [13] of 5000 images on class conditional ImageNet with resolution 512×512 (please see more implementation details in Appendix D), shown in Appendix Fig. 12. We present more visualization results in Appendix Fig. 23.

We further study samples synthesized with and without NPNet to assess the distribution shift. Our empirical analysis reveals that NPNet effectively mitigates distributional discrepancies, resulting in synthesized images that exhibit statistically significant alignment with the true data manifold, shown in Appendix Fig. 13.

Table 4. Ablation studies of the proposed methods on SDXL.

Method	PickScore (↑)	HPSv2 (↑)	AES (↑)	ImageReward (↑)
Standard	21.69	28.48	6.0373	58.01
NPNet w/o singular value prediction	21.49	27.76	6.0164	49.03
NPNet w/o residual prediction	21.83	28.55	6.0315	63.05
NPNet w/o data selection	21.73	28.46	6.0375	62.91
NPNet	21.86	28.68	6.0540	65.01

4.2. Analysis of Generalization and Robustness

To validate the superior generalization ability of our NPNet, we conduct multiple experiments, covering experiments of cross-dataset, cross-model architecture, various inference steps, various random seed ranges, stochastic/deterministic samplers, and the integration of other methods.

Generalization to Various Datasets and Models. Although we trained our NPNet exclusively with the NPDs collected from the Pick-a-Pic dataset, the experimental results presented in Table 2 and Table 3, and Fig. 3 demonstrate that our model exhibits strong cross-dataset generalization capabilities, achieving impressive results on other datasets as well. In addition to the fine-tuning experiments detailed in Table 3, we also applied the NPNet trained on NPD collected from SDXL to DreamShaper-xl-v2-turbo, evaluating the performance with our NPNet without any

fine-tuning. The experiment results are shown in Appendix Table 6. Moreover, We use the same NPNet from SDXL and apply it directly on few steps T2I diffusion model, such as LCM [33], PCM [51] and SDXL-Lightning [26], which also improves the quality of the synthesized images, shown in Appendix Table 18. These results indicate promising performance with our NPNet, underscoring the model’s capability for cross-model generalization. We also conduct experiments on noise seed range in Appendix Table 19, the results demonstrate that our NPNet exhibits strong generalization capabilities across the out-of-distribution random seed ranges without harming the image diversity.

Generalization to Stochastic/Deterministic Samplers.

When collecting NPD on SDXL, we use the deterministic DDIM sampler. However, whether the NPNet can effectively perform with stochastic samplers is crucial. To investigate our NPNet’s performance across various sampling methods, we evaluated 7 different samplers, using NPNet trained on the NPD collected from SDXL, whose sampler is DDIM. The results shown in Fig. 5 and Appendix Table 11 suggest that our NPNet is adaptable and capable of maintaining high performance even when subjected to various levels of randomness in the sampling process, further validating the generalization of our NPNet.

Table 5. We conducted combinatorial experiments with other mainstream methods that can improve the alignment between the text prompts and synthesized images. The results indicate that our approach is orthogonal to these methods, allowing for joint usage to achieve improved performance.

Methods	PickScore (↑)	HPSv2 (↑)	AES (↑)	ImageReward (↑)
Standard	21.07	25.38	5.9058	34.04
DPO	21.83	28.42	5.9993	68.82
DPO+NPNet (ours)	21.91	28.70	6.0277	75.06
Standard	21.07	25.38	5.9058	34.04
AYS	21.53	27.24	6.0310	50.74
AYS+NPNet (ours)	21.76	28.14	6.1239	59.50

Orthogonal Experiment. To explore whether our model can be combined with other approaches, which aim at enhancing the semantic faithfulness of synthesized images, such as DPO [40] and AYS [43], we conduct combination experiments, shown in Table 5. Note that AYS only releases the code under the inference step 10, so we conduct the combinatorial experiment with AYS and DPO (for consistency) under the inference step 10. The results indicate that our method is orthogonal to these works, allowing for joint usage to further improve the quality of synthesized images.

More notably, we significantly improve the performance and winning rate of the previous noise optimization methods by using NPNet together with them, which further illustrates the effectiveness and scalability of our NPNet, shown in Appendix Table 20 and Fig. 19.

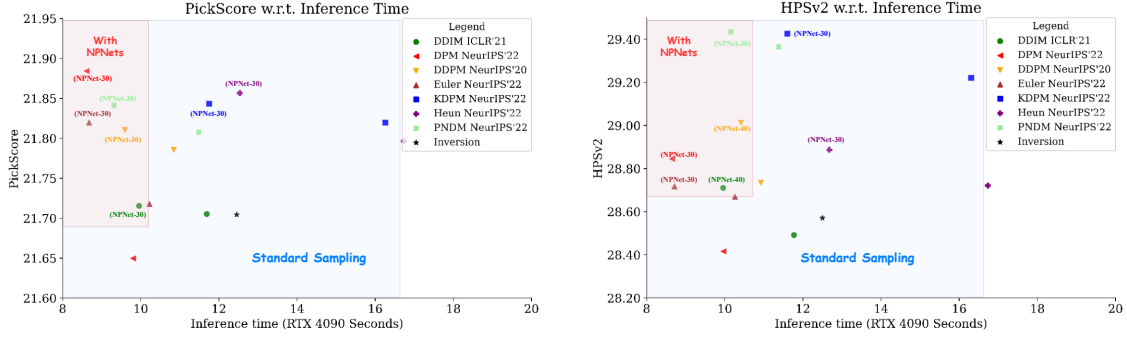


Figure 5. We evaluate our NPNet with 7 samplers on SDXL in Pick-a-Pic dataset, including both the deterministic sampler and stochastic sampler (with a default inference step 50). “NPNet-30” means the inference step is 30 with NPNet. The red area in the top left corner of the image represents the results of efficient high-performance methods, while the experimental results of NPNet are nearly in that same region. It highlights that NPNet is capable of synthesizing higher-quality images with fewer steps and consuming less time. Moreover, the results demonstrate the generalization ability of our NPNet across different samplers.

Robustness to the Hyper-parameters. We study how the performance of NPNet is robust to the hyper-parameters. We first evaluate the performance of our NPNet under various inference steps, as illustrated in Fig. 4, Appendix Fig. 14, Appendix Fig. 16, and Appendix Fig. 17. These results highlight the generalization and versatility of our NPNet is robust to the common range of inference steps. Such consistency suggests that the model is well-tuned to adapt to different conditions, making it effective for a wide range of applications. We also do exploration studies on the other hyper-parameters, such as batch size, the training epochs, and CFG values in Appendix Table 12. The studied optimal settings are the batch size with 64, the training epochs with 30, and the CFG value with 5.5. Moreover, we explore the influences of different amounts of training samples, shown in Appendix Table 17.

Ablation Studies. We conduct ablation studies about the architecture designs of NPNet in Table 4. The results show that both *singular value prediction* and *residual prediction* contribute to the final optimal results, while the *singular value prediction* component plays a more important role. We also empirically verify the effectiveness of data selection strategies in Appendix Tables 9 and 10.

4.3. Efficiency Analysis and Downstream Tasks

Efficiency Analysis. As a plug-and-play module, it is essential to discuss the memory consumption and inference latency of NPNet. Remarkably, NPNet achieves significant performance improvements even with fewer inference steps and reduced time costs in Fig. 5. Even when operating at the same number of inference steps in Appendix Table 11, our model introduces only a 0.4-second delay while synthesizing high-quality images, demonstrating its efficiency. Additionally, Appendix Fig. 15 shows its memory consumption is mere 500 MB, highlighting its resource-friendly design.

Our model not only delivers superior results but also exhibit significant application potential and practical value due to the impressive deployment efficiency.

Exploration of Downstream Task. We explored the potential of integrating NPNet with downstream tasks, specifically by combining it with ControlNet [58] for controlled image synthesis. As a plug-and-play module, our NPNet can be seamlessly incorporated into ControlNet. Visualization results in Appendix Fig. 20 demonstrate that this integration leads to the synthesis of more detailed and higher-quality images with higher semantic faithfulness, highlighting the effectiveness of our approach.

5. Conclusion

In this paper, we introduce a novel concept termed the *noise prompt*, which aims to transform random Gaussian noise into a golden noise by incorporating a small perturbation derived from the text prompt. Building upon this, we firstly formulate a *noise prompt learning* framework that systematically obtains “prompted” golden noise associated with a text prompt for diffusion models, by constructing a *noise prompt dataset* collection pipeline that incorporates HPSv2 to filter our data and designing several backbones for our *noise prompt models*. Our extensive experiments demonstrate the superiority of NPNet, which is plug-and-play, straightforward, and nearly time-efficient, while delivering significant performance improvements. We believe that the future application of NPNet will be broad and impactful, encompassing video, 3D content, and seamless deployment of AIGC products, thereby making a meaningful contribution to the community.

Acknowledgement

This work was supported by the Science and Technology Bureau of Nansha District Under Key Field Science and Technology Plan Program No. 2024ZD002. This work was supported by Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things (No.2023B1212010007).

References

- [1] Donghoon Ahn. Self-rectifying diffusion sampling with perturbed-attention guidance, 2024. 27
- [2] Donghoon Ahn, Jiwon Kang, Sanghyun Lee, Jaewon Min, Minjae Kim, Wooseok Jang, Hyoungwon Cho, Sayak Paul, SeonHwa Kim, Eunju Cha, Kyong Hwan Jin, and Seungryong Kim. A noise is worth diffusion guidance, 2024. 2
- [3] Shane Barratt and Rishi Sharma. A note on the inception score, 2018. 12
- [4] James Betker, Gabriel Goh, Li Jing, † TimBrooks, Jianfeng Wang, Linjie Li, † LongOuyang, † JuntangZhuang, † JoyceLee, † YufeiGuo, † WesamManassra, † PrafullaDhariwal, † CaseyChu, † YunxinJiao, and Aditya Ramesh. Improving image generation with better captions. 2
- [5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models, 2023. 2, 14, 15
- [6] Hao Chen, Yujin Han, Diganta Misra, Xiang Li, Kai Hu, Difan Zou, Masashi Sugiyama, Jindong Wang, and Bhiksha Raj. Slight corruption in pre-training data makes better diffusion models, 2024. 13
- [7] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- α : Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. 2
- [8] Sherry X. Chen, Yaron Vaxman, Elad Ben Baruch, David Asulin, Aviad Moreschet, Kuo-Chin Lien, Misha Sra, and Pradeep Sen. Tino-edit: Timestep and noise optimization for robust diffusion-based image editing, 2024. 2
- [9] Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. Cfg++: Manifold-constrained classifier free guidance for diffusion models, 2024. 27
- [10] Dhruva Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment, 2023. 5
- [11] Xiefan Guo, Jinlin Liu, Miaomiao Cui, Jiankai Li, Hongyu Yang, and Di Huang. Initno: Boosting text-to-image diffusion models via initial noise optimization, 2024. 2, 14, 15
- [12] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022. 2, 12
- [13] 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, 2018. 7, 11
- [14] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *Neural Information Processing Systems Workshop*, Virtual Event, 2021. NeurIPS. 3
- [15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Neural Information Processing Systems*, pages 6840–6851, Virtual Event, 2020. NeurIPS. 19
- [16] Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 15, 27
- [17] Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention, 2024. 15
- [18] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7462–7471, 2023. 27
- [19] Minghui Hu, Jianbin Zheng, Chuanxia Zheng, Chaoyue Wang, Dacheng Tao, and Tat-Jen Cham. One more step: A versatile plug-and-play module for rectifying diffusion schedule flaws and enhancing low-frequency controls, 2023. 13
- [20] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xi-hui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *arXiv preprint arXiv:2307.06350*, 2023. 5
- [21] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. 22
- [22] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *ArXiv*, abs/2206.00364, 2022. 19
- [23] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation, 2023. 2, 5, 11
- [24] Kolors. Kolors: Effective training of diffusion model for photorealistic text-to-image synthesis. *arXiv preprint*, 2024. 2
- [25] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xincheng Deng, and Yingfang Zhang. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding, 2024. 2, 13
- [26] Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation, 2024. 7, 22
- [27] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 11
- [28] Buhua Liu, Shitong Shao, Bao Li, Lichen Bai, Zhiqiang Xu, Haoyi Xiong, James Kwok, Sumi Helal, and Zeke Xie. Alignment of diffusion models: Fundamentals, challenges, and future, 2024. 2

- [29] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds, 2022. [19](#)
- [30] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, 2021. [3](#)
- [31] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *Neural Information Processing Systems*, New Orleans, LA, USA, 2022. NeurIPS. [19](#)
- [32] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models, 2022. [2](#), [13](#), [27](#)
- [33] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2023. [7](#), [22](#)
- [34] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models, 2023. [4](#), [13](#)
- [35] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. [2](#)
- [36] Pablo Pernias, Dominic Rampas, Mats L. Richter, Christopher J. Pal, and Marc Aubreville. Wuerstchen: An efficient architecture for large-scale text-to-image diffusion models, 2023. [2](#)
- [37] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2023. [2](#), [13](#)
- [38] Zipeng Qi, Lichen Bai, Haoyi Xiong, and Zeke Xie. Not all noises are created equally: diffusion noise selection and optimization, 2024. [2](#), [14](#)
- [39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. [12](#)
- [40] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. [7](#)
- [41] 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 (CVPR)*, pages 10684–10695, 2022. [14](#), [15](#)
- [42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Computer Vision and Pattern Recognition*, pages 10684–10695. IEEE, 2022. [2](#)
- [43] Amirmojtaba Sabour, Sanja Fidler, and Karsten Kreis. Align your steps: Optimizing sampling schedules in diffusion models, 2024. [7](#)
- [44] Seyedmorteza Sadat, Otmar Hilliges, and Romann M. Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models, 2024. [27](#)
- [45] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. [5](#)
- [46] Christoph Schuhmann. Improved aesthetic predictor. [2](#), [12](#)
- [47] Chenyang Si, Ziqi Huang, Yuming Jiang, and Ziwei Liu. Freeu: Free lunch in diffusion u-net, 2023. [15](#), [27](#)
- [48] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, kigali, rwanda, 2023. OpenReview.net. [3](#), [19](#)
- [49] GW Stewart. Matrix perturbation theory. *Computer Science and Scientific Computing/Academic Press, Inc*, 1990. [4](#)
- [50] Michael Toker, Hadas Orgad, Mor Ventura, Dana Arad, and Yonatan Belinkov. Diffusion lens: Interpreting text encoders in text-to-image pipelines, 2024. [2](#)
- [51] Fu-Yun Wang, Zhaoyang Huang, Alexander William Bergman, Dazhong Shen, Peng Gao, Michael Lingelbach, Keqiang Sun, Weikang Bian, Guanglu Song, Yu Liu, et al. Phased consistency model. *arXiv preprint arXiv:2405.18407*, 2024. [7](#), [22](#)
- [52] Qiucheng Wu, Yujian Liu, Handong Zhao, Ajinkya Kale, Trung Bui, Tong Yu, Zhe Lin, Yang Zhang, and Shiyu Chang. Uncovering the disentanglement capability in text-to-image diffusion models, 2022. [2](#)
- [53] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis, 2023. [2](#), [5](#), [11](#)
- [54] Zeke Xie, Qianyu Tang, Mingming Sun, and Ping Li. On the overlooked structure of stochastic gradients. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [4](#)
- [55] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. [2](#), [12](#)
- [56] Katherine Xu, Lingzhi Zhang, and Jianbo Shi. Good seed makes a good crop: Discovering secret seeds in text-to-image diffusion models, 2024. [2](#)
- [57] Hu Yu, Hao Luo, Fan Wang, and Feng Zhao. Uncovering the text embedding in text-to-image diffusion models, 2024. [2](#)
- [58] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. [8](#)
- [59] Sixian Zhang, Bohan Wang, Junqiang Wu, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Learning multi-dimensional human preference for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8018–8027, 2024. [2](#), [12](#)