
Invertible Consistency Distillation for Text-Guided Image Editing in Around 7 Steps

Nikita Starodubcev^{1 2} Mikhail Khoroshikh^{1 2} Artem Babenko¹ Dmitry Baranchuk¹

¹Yandex Research

²HSE University

<https://yandex-research.github.io/invertible-cd>



Figure 1: Invertible Consistency Distillation (iCD) enables both fast image editing and strong generation performance in a few model evaluations.

Abstract

Diffusion distillation represents a highly promising direction for achieving faithful text-to-image generation in a few sampling steps. However, despite recent successes, existing distilled models still do not provide the full spectrum of diffusion abilities, such as real image inversion, which enables many precise image manipulation methods. This work aims to enrich distilled text-to-image diffusion models with the ability to effectively encode real images into their latent space. To this end, we introduce *invertible Consistency Distillation* (iCD), a generalized consistency distillation framework that facilitates both high-quality image synthesis and accurate image encoding in only 3–4 inference steps. Though the inversion problem for text-to-image diffusion models gets exacerbated by high classifier-free guidance scales, we notice that *dynamic guidance* significantly reduces reconstruction errors without noticeable degradation in generation performance. As a result, we demonstrate that iCD equipped with dynamic guidance may serve as a highly effective tool for zero-shot text-guided image editing, competing with more expensive state-of-the-art alternatives.

1 Introduction

Recently, text-to-image diffusion models [1, 2, 3, 4, 5, 6] have become a dominant paradigm in image generation based on user-provided textual prompts. The exceptional quality of these models makes them a valuable tool for graphics editors, especially for various image manipulation tasks [7, 8, 9]. In practice, however, the applicability of diffusion models is often hindered by their slow inference, which stems from a sequential sampling procedure, gradually recovering images from pure noise.

To speed-up the inference, many recent works aim to reduce the number of diffusion steps via diffusion distillation [10, 11, 12, 13, 14, 15, 16, 17] that has provided significant progress in high-quality generation in 1–4 steps and has already been successfully scaled to the state-of-the-art text-to-image diffusion models [18, 19, 20, 21, 22, 23, 24]. Though the existing distillation approaches still often trade either mode coverage or image quality for few-step inference, the proposed models can already be feasible for practical applications, such as text-driven image editing [25, 26, 27].

The most effective diffusion-based editing methods typically require encoding real images into the latent space of a diffusion model. For “undistilled” models, this encoding is possible by virtue of the connection of diffusion modeling [28] with denoising score matching [29] through SDE and probability flow ODE (PF ODE) [30]. The ODE perspective of diffusion models reveals their *reversibility*, i.e., the ability to encode a real image into the model latent space and closely reconstruct it with minimal changes. This ability is successfully exploited in various applications, such as text-driven image editing [31, 32, 33], domain translation [34, 9], style transfer [35].

Nevertheless, it remains unclear if distilled models can be enriched with such reversibility since existing diffusion distillation methods primarily focus on achieving efficient generation. This work positively answers this question by proposing *invertible Consistency Distillation* (iCD), a generalized consistency modeling framework [10, 12, 13] enabling both high-quality image generation and accurate inversion in a few sampling steps.

In practice, text-to-image models leverage classifier-free guidance (CFG) [36], which is crucial for high-fidelity text-to-image generation [1, 3] and text-guided editing [32, 33]. However, the guided diffusion processes yield significant challenges for inversion-based editing methods [32]. Previous approaches [32, 37, 38, 39, 40, 41, 42, 27, 25, 26, 43, 44] have extensively addressed these challenges but often necessitate high computational budget to achieve both strong image manipulations and faithful content preservation. While some of these techniques are applicable to the distilled models [25, 26, 27], they still dilute the primary advantage of distilled diffusion models: efficient inference.

One of the main ingredients of the iCD framework is how it operates with guided diffusion processes. Recently, dynamic guidance has been proposed to improve distribution diversity without noticeable loss in image quality [45, 46]. The key idea is to deactivate CFG for high diffusion noise levels to stimulate exploration at earlier sampling steps. In this work, we notice that dynamic CFG can facilitate image inversion while preserving the editability of the text-to-image diffusion models. Notably, dynamic CFG yields no computational overhead, entirely leveraging the efficiency gains from diffusion distillation. In our experiments, we demonstrate that invertible distilled models equipped with dynamic guidance are a highly effective inversion-based image editing tool.

To sum up, our contributions can be formulated as follows:

- We propose a generalized consistency distillation framework, invertible Consistency Distillation (iCD), enabling both high-fidelity text-to-image generation and accurate image encoding in around 3–4 sampling steps.
- We investigate dynamic classifier-free guidance in the context of image inversion and text-guided editing. We demonstrate that it preserves editability of the text-to-image diffusion models while significantly increasing the inversion quality for free.
- We apply iCD to large-scale text-to-image models such Stable Diffusion 1.5 [4] and XL [1] and extensively evaluate them for image editing problems. According to automated and human studies, we confirm that iCD unlocks faithful text-guided image editing for 6–8 steps and is comparable to state-of-the-art text-driven image manipulation methods while being multiple times faster.

2 Background

Diffusion probabilistic models DPMs [29, 28, 47] are a class of generative models producing samples from a simple, typically standard normal, distribution by solving the underlying Probability Flow ODE [30, 48], involving iterative *score function* estimation. DPMs are trained to approximate the score function and employ dedicated diffusion ODE solvers [49, 50, 48] for sampling. DDIM [49] is a simple yet effective solver, widely used in text-to-image models and operating in around 50 steps. A single DDIM step from \mathbf{x}_t to \mathbf{x}_s can be formulated as follows:

$$\mathbf{x}_s^w = \text{DDIM}(\mathbf{x}_t, t, s, \mathbf{c}, w) = \sqrt{\frac{\alpha_s}{\alpha_t}} \mathbf{x}_t + \epsilon_{\theta}^w(\mathbf{x}_t, t, \mathbf{c}) \left(\sqrt{1 - \alpha_s} - \sqrt{\frac{\alpha_s}{\alpha_t}} \sqrt{1 - \alpha_t} \right), \quad (1)$$

where α_s, α_t are defined according to the diffusion schedule [47], and $\epsilon_{\theta}^w(\mathbf{x}_t, t, \mathbf{c}) = \epsilon_{\theta}(\mathbf{x}_t, t, \emptyset) + w(\epsilon_{\theta}(\mathbf{x}_t, t, \mathbf{c}) - \epsilon_{\theta}(\mathbf{x}_t, t, \emptyset))$ is a linear combination of conditional and unconditional noise predictions called as Classifier-Free Guidance (CFG) [51], used to improve the image quality and context alignment in conditional generation. In the following, we omit the condition \mathbf{c} for simplicity.

Due to reversibility, the PF ODE can be solved in both directions: encoding data into the noise space and decoding it back without additional optimization procedures. We refer to this process as *inversion* [32] where encoding and decoding correspond to *forward* and *reverse* processes, respectively.

Consistency Distillation CD [10, 12, 13] is the recent state-of-the-art diffusion distillation approach for few-step image generation, which learns to integrate the PF ODE induced with a pretrained diffusion model. In more detail, the model \mathbf{f}_{θ} is trained to satisfy the self-consistency property:

$$\mathcal{L}_{\text{CD}}(\theta) = \mathbb{E} \left[d(\mathbf{f}_{\theta}(\mathbf{x}_{t_{n-1}}^w, t_{n-1}), \mathbf{f}_{\theta}(\mathbf{x}_{t_n}, t_n)) \right] \rightarrow \min_{\theta}, \quad (2)$$

where $t_n \in \{t_0, \dots, t_N\}$ is a discrete time step, $d(\cdot, \cdot)$ denotes a distance function and $\mathbf{x}_{t_{n-1}}^w$ is obtained with a single step of the DDIM solver from t_n to t_{n-1} using the teacher diffusion model. The optimum of (2) is defined by the boundary condition, $\mathbf{f}_{\theta}(\mathbf{x}_{t_0}, t_0) = \mathbf{x}_{t_0}$. Therefore, consistency models (CMs) learn the transition from any trajectory point to the starting one: $\mathbf{f}_{\theta}(\mathbf{x}_{t_n}, t_n) = \mathbf{x}_{t_0}, \forall t_n \in \{t_0, \dots, t_N\}$. Consequently, CMs imply a single step generation. However, approximating the entire trajectory using only one step remains highly challenging, leading to unsatisfactory results in practice. To address this, [10] proposes stochastic *multistep consistency sampling* that iteratively predicts \mathbf{x}_{t_0} using \mathbf{f}_{θ} and goes back to the intermediate points using the forward diffusion process.

The competitive performance of consistency models has stimulated their rapid adoption for text-to-image generation [52, 18, 53]. Nevertheless, we believe that CMs have not yet fully realized their potential in downstream applications, where DPMs excel. One of the reasons is that, unlike DPMs, CMs do not support the inversion process. This work aims to unlock this ability for CMs.

Dynamic guidance State-of-the-art text-to-image models employ large CFG scales to achieve high image quality and textual alignment. However, it often leads to the reduced diversity of generated images. To address this, *dynamic classifier-free guidance* [45, 46, 54] has recently been proposed to improve distribution diversity without noticeable loss in generative performance. CADS [45] gradually increases the guidance scale from zero to the initial high value over the sampling process, Figure 2a. Alternatively, [46] proposes deactivating the guidance for low and high noise levels and using it only on the middle time step interval, Figure 2b. Both strategies suggest that the unguided process at high noise levels is responsible for better distribution diversity without compromising sampling quality. In addition, the authors [46] demonstrate that guidance at low noise levels has a minor effect on the performance and can be omitted to avoid extra model evaluations for guidance calculation. Both dynamic techniques are controlled by two hyperparameters: τ_1 and τ_2 , which are responsible for the value of dynamic CFG $w(t)$. In our work, we focus on the CADS formulation.

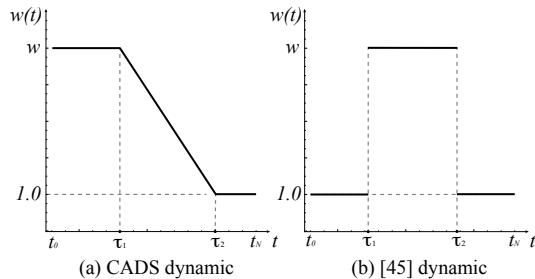


Figure 2: Dynamic CFG strategies.

3 Method

This section introduces the invertible Consistency Distillation (iCD) framework, which comprises forward and reverse consistency models. First, we formulate the forward CD procedure that encodes images into latent noise. Then, we describe multi-boundary generalization of iCD to enable deterministic multistep inversion. Finally, we investigate the *dynamic guidance* technique from the inversion perspective.

3.1 Forward Consistency Distillation

Forward Consistency Distillation (fCD) works in the opposite way to CD. That is, it aims to map any point on the PF ODE trajectory to the latent noise (the last trajectory point).

The transition from CD to the forward counterpart is quite straightforward: the only thing that should be modified is the boundary condition. Precisely, the forward consistency model is constrained to be an identity function for the last trajectory point: $f_\theta(\mathbf{x}_{t_N}, t_N) = \mathbf{x}_{t_N}$. Thus, fCD inherits the same consistency distillation loss (2) without incurring extra training costs. This way, the distilled model can transform any trajectory point to the last one: $f_\theta(\mathbf{x}_{t_n}, t_n) = \mathbf{x}_{t_N}, \forall t_n$. To perform inversion, first, fCD encodes an image into noise and then CD decodes it back. The comparison between CD and fCD is shown in Figure 3a.

3.2 Multi-boundary Consistency Distillation

In practice, the encoding with fCD faces two challenges. Firstly, like in CD, a single-step prediction with fCD can be highly inaccurate. However, this cannot be easily addressed since the multistep consistency sampling [10] is not applicable to fCD. Concretely, intermediate points cannot be obtained from the latent noise using the forward diffusion process. Secondly, even if fCD is accurate, the multistep sampling is not suitable for decoding, as its stochastic nature prevents the reconstruction of real images. So, to improve the prediction accuracy of fCD and reduce the reconstruction error of CD, it is necessary to formulate a deterministic multistep procedure for both models.

Recent approaches [53, 13] generalize the CD framework to a multistep regime and allow approximating arbitrary trajectory intervals in the reverse direction. However, these methods focus solely on the generation quality, without supporting the inversion. Thus, inspired by these works, we propose a multi-boundary CD, that unlocks deterministic multistep inversion with the distilled models and carries similar training costs as the classical CD methods.

Specifically, we divide the solution interval, $\{t_0, \dots, t_N\}$, into m segments and perform the distillation on each of these segments separately. This way, we obtain a set of single-step consistency models operating on different intervals and boundary points. This formulation is valid for both CD and fCD and hence can enable deterministic multistep inversion. We provide an illustration of 2-boundary CD and fCD in Figure 3b. We denote the multi-boundary reverse and forward models as CD^m and fCD^m .

Formally, we consider CD^m and fCD^m using the following parametrization, inspired by [53, 13].

$$\mathbf{x}_{s_t^m} = f_\theta^m(\mathbf{x}_t, t, s_t^m, w) = DDIM(\mathbf{x}_t, t, s_t^m, w), \quad (3)$$

where s_t^m is the boundary time step depending on the number of boundaries, m , and the current time step t . For instance, let $m = 1$, then $s_t^1 = t_0$ for CD^1 and $s_t^1 = t_N$ for fCD^1 . Note that we learn a single model, the multistep sampling is achieved by varying s_t^m during inference. The training objective remains the same as (2), avoiding additional training costs compared to CD. The only limitation is that the number of segments and the corresponding boundary time steps must be set before the training.

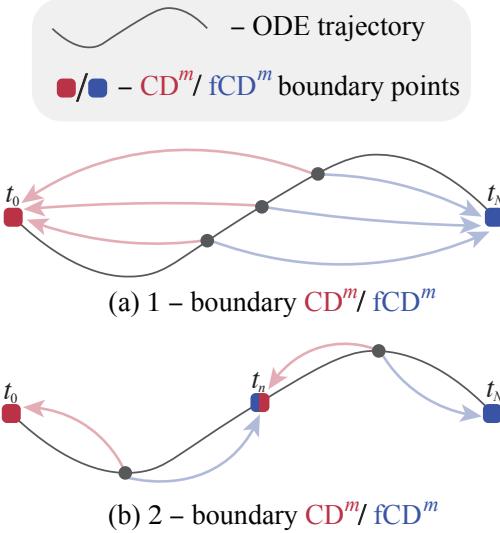


Figure 3: The proposed invertible Consistency Distillation framework consists of two models: the forward m -boundary model, fCD^m , and the reverse model, CD^m . (a) For $m = 1$, the reverse model corresponds to CD. More boundary points unlock the deterministic multistep inversion, e.g., (b) shows the case for $m = 2$.

3.3 Training fCD^m and CD^m

We train fCD^m and CD^m separately, initializing both with the same teacher model. We use the same loss with a difference only in boundary time steps. However, a notable difference is the CFG scale, w . For CD^m, we preliminarily embed the model on guidance, following [20], to use various w during sampling and avoid extra model evaluations. For fCD^m, we consider an unguided model with a constant $w = 1$. The reason is that the guided encoding ($w > 1$) leads to out-of-distribution latent noise [32], and as a result, to poor image reconstruction. We confirm this intuition in Section 3.4. Finally, we find that $m=3-4$ provides competitive generation and inversion quality for large-scale text-to-image models [4, 1].

Preservation losses The procedure described above already provides decent inversion quality but still does not match the teacher inversion performance. To reduce the gap between them, we propose the forward and reverse preservation losses aimed at making CD^m and fCD^m more consistent with each other and improve the inversion accuracy. These losses can additionally be turned on during training. Below, we denote the parameters of CD^m, fCD^m as θ^+ , θ^- , respectively.

The forward preservation loss modifies only fCD^m and is described as follows:

$$\mathcal{L}_f(\theta^-, \theta^+) = \mathbb{E} [d(f_{\theta^-}^m(f_{\theta^+}^m(\mathbf{x}_{s_t^m})), \mathbf{x}_{s_t^m})] \rightarrow \min_{\theta^-}, \quad (4)$$

For simplicity, we omit some notation. In a nutshell, we sample a noisy image $\mathbf{x}_{s_t^m}$ for a boundary time step s_t^m , then make a prediction using CD^m and force fCD^m to predict the same $\mathbf{x}_{s_t^m}$. This approach encourages CD^m and fCD^m to be consistent with each other.

The reverse preservation loss provides the same intuition but with a difference in the optimized model (CD^m instead of fCD^m) and prediction sequence. That is, we first make a prediction using fCD^m and then use CD^m. We denote it as $\mathcal{L}_r(\theta^-, \theta^+)$. In our experiments, we calculate the preservation losses only for the unguided reverse process ($w = 1$).

Putting it all together We present our final pipeline for the case where fCD^m and CD^m are trained jointly starting from the pretrained diffusion model. However, it is possible to learn them by one or take an already pretrained consistency model and learn the rest one. The final objective consists of two consistency losses with the proposed multi-boundary modification and two preservation losses:

$$\mathcal{L}_{iCD}(\theta^+, \theta^-) = \mathcal{L}_{CD}(\theta^+) + \mathcal{L}_{CD}(\theta^-) + \lambda_f \mathcal{L}_f(\theta^-, \theta^+) + \lambda_r \mathcal{L}_r(\theta^+, \theta^-) \quad (5)$$

In this way, the proposed approach can compete with the state-of-the-art inversion methods using heavyweight diffusion models. We present technical details about the training in Appendix A.

3.4 Dynamic Classifier-Free Guidance Facilitates Inversion

As previously discussed, dynamic guidance [45, 46] provides promising results for both faithful and diverse text-to-image generation. In this work, we reveal that dynamic CFG is also an effective technique for improving inversion accuracy as shown in Figure 4b. Below, we delve into the questions when and why dynamic guidance might facilitate image inversion while preserving the generative performance. To answer these questions, we conduct experiments using Stable Diffusion 1.5 with DDIM solver for 50 steps and maximum CFG scale set to 8.0.

Dynamic guidance for decoding We start with the dynamic CFG analysis at the decoding stage using the unguided encoding process following the prior work [32]. First, we wonder at which time steps the guidance has the most significant impact on reconstruction quality. To this end, we evaluate MSE between real and reconstructed images for different CFG turn-on thresholds T . If $t > T$, we set $w = 1.0$, otherwise, the CFG scale is set to its initial value 8.0. In Figure 4a, we observe an exponential decrease in reconstruction error, implying that the absence of CFG at higher noise levels is essential for achieving more accurate inversion. Figure 4b confirms this intuition qualitatively. These results are consistent with [45, 46], which also suggest turning off the guidance at high noise levels but motivating this from the perspective of diversity improvement.

Then, we investigate the influence of various τ_1, τ_2 from the CADS dynamic (Figure 2a) on the inversion and generation performance. We aim to identify an operating point providing both strong

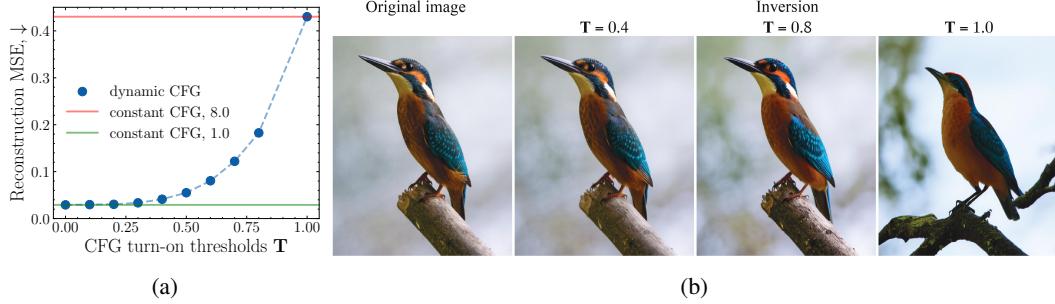


Figure 4: (a) Reconstruction error of the decoding process for different CFG turn-on thresholds. (b) Image inversion examples for different CFG turn-on thresholds T . Guidance at high noise levels ($T = 1.0$) drastically degrades the inversion quality.

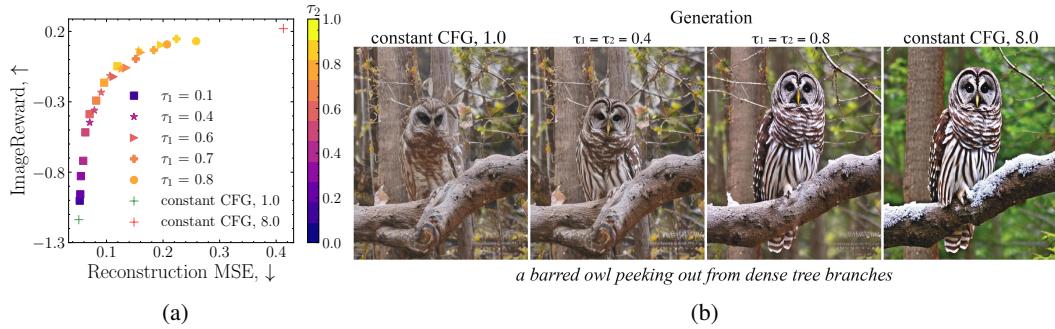


Figure 5: (a) Trade-off between generation performance (IR) and reconstruction quality (MSE) provided by different τ_1, τ_2 . (b) Generation examples for dynamic and constant CFG scales. The points around $\tau_1 = \tau_2 = 0.8$ provide preferable trade-off between generation and inversion performance.

	Enc, No CFG	Enc, d.CFG, $\tau=0.6$	Enc, d.CFG, $\tau=0.8$	Enc, CFG
Dec. No CFG	11.0	36.6	63.4	100.5
Dec. d.CFG $\tau=0.6$	11.4	23.5	67.8	102.6
Dec. d.CFG $\tau=0.8$	15.0	14.6	52.2	108.5
Dec. CFG	19.0	19.2	19.0	102.1
Latent NLL, \downarrow	1.401	1.409	1.415	1.428

Table 1: FID-5k for SD1.5 starting from the noise latents obtained using different encoding strategies, and NLL for these latents. Though encoding with dynamic CFG produces consistently more plausible latents than constant CFG, the unguided encoding remains preferable.

generation performance and faithful image inversion. Thus, we evaluate generation performance using the ImageReward [55] (IR) on top of randomly generated samples for 1000 COCO2014 prompts [56]. The inversion accuracy is estimated in terms of MSE between original and reconstructed samples. Figure 5a presents the results for varying τ_1 and τ_2 . It can be seen that several points for $\tau_1 \geq 0.7$ offer slightly lower text-to-image performance but exhibit significantly better reconstruction quality compared to the constant CFG scale, 8.0. Moreover, we notice that the settings where $\tau_1 = \tau_2$ perform similarly to those where $\tau_1 < \tau_2$. Consequently, in all our experiments, we consider a single τ representing the case where $\tau_1 = \tau_2$ and use $\tau = 0.7$ and $\tau = 0.8$. This means that CD^m follows unguided sampling for $t > \tau$ and sets the initial CFG scale for $t \leq \tau$.

Note that the setting with $\tau = \tau_1 = \tau_2$ corresponds to a step CFG function $w(t)$, which yields a distinct advantage for distilled models. The linearly changing CFG scales are not applicable to the processes with large discretization steps, typical for distilled diffusion models. Therefore, such a CFG schedule needs to be distilled into the model during training, making it less flexible for different generation and editing settings. In contrast, the step CFG function enables dynamic CFG for already pretrained distilled models, operating with different constant CFG scales.

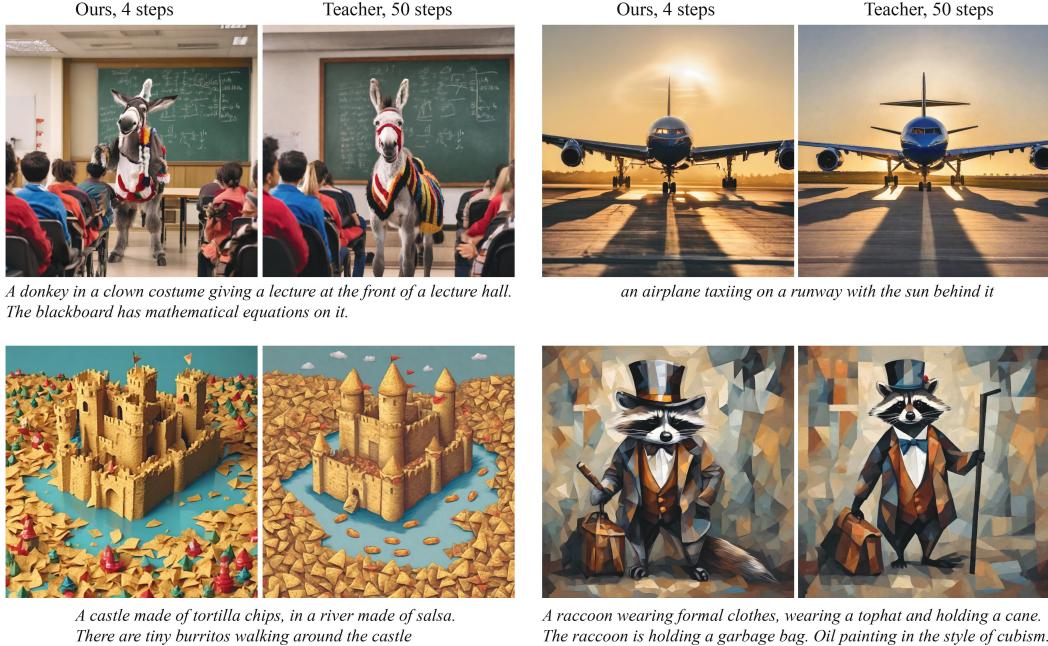


Figure 6: Few examples of text-to-image generation using the iCD-XL model for 4 steps.

Dynamic guidance for encoding Next, we investigate the guidance role in tackling the encoding problem. In Table 1, we compare noise latents encoded using various guidance strategies. The quality of the noise latents is estimated by evaluating the generation performance starting from these latents with a fixed CFG scale of 8.0. As the performance measure, we calculate FID for 5000 image-text pairs from COCO [56].

We observe that the latents obtained with a consistently high CFG scale exhibit the worst generative performance, indicating their out-of-domain nature. While dynamic guidance produces significantly more plausible latents, it still falls short of the unguided encoding in most cases. To further validate these results, we estimate the negative log-likelihood (NLL) of the encoded latents under different CFG settings in Table 1 (Bottom). NLL is calculated with respect to the standard normal distribution. While NLL decreases for dynamic CFG with lower τ , the encoding without guidance ($w=1$) provides the highest likelihood value. Therefore, in all our experiments, we maintain $w=1$ for the encoding and train the forward distilled models (fCD) on the unguided teacher process.

4 Experiments

In the following experiments, we apply our approach to text-to-image diffusion models of different scales: SD1.5 [57] and SDXL [1], and denote them iCD-SD1.5 and iCD-XL, respectively. We provide the training details in Appendix A.

Initially, we illustrate the inversion capability of the proposed framework. Then, we consider the text-guided image editing problem and demonstrate that our approach outperforms or is comparable to significantly more expensive baselines.

Before diving into the main experiments, we present a few generated samples using iCD-XL for 4 steps in Figure 6. Additional quantitative and qualitative results are provided in Appendix B. The results confirm that the distilled model demonstrates strong text-to-image generation performance.

4.1 Inversion quality of iCD

Here, we analyze the reconstruction capabilities of iCD-SD1.5 under various configurations. Specifically, we explore the contribution of the different pipeline components, such as the number of steps, preservation losses, and dynamic CFG, to inversion performance. Our forward model is run without

Configuration	LPIPS ↓	DinoV2 ↑	PSNR ↑
Unguided decoding setting			
fDDIM ⁵⁰ DDIM ⁵⁰	0.167	0.834	22.98
A fCD ² CD ²	0.332	0.632	17.75
B fCD ³ CD ³	0.317	0.649	18.42
C fCD ⁴ CD ⁴	0.276	0.715	19.19
E fCD ⁴ + \mathcal{L}_t CD ⁴	0.484	0.554	16.40
F fCD ⁴ + \mathcal{L}_t CD ⁴	0.248	0.728	20.01
G fCD ⁴ + \mathcal{L}_t CD ⁴ + \mathcal{L}_r	0.198	0.837	22.27
Guided decoding setting, $w = 8$			
fDDIM ⁵⁰ DDIM ⁵⁰	0.479	0.534	14.12
fDDIM ⁵⁰ DDIM ⁵⁰ + d.CFG	0.279	0.726	19.58
H fCD ⁴ CD ⁴	0.476	0.550	13.87
I fCD ⁴ CD ⁴ + d.CFG	0.370	0.650	16.72
J fCD ⁴ + \mathcal{L}_t CD ⁴ + d.CFG	0.317	0.698	17.98
K fCD ⁴ + \mathcal{L}_t CD ⁴ + \mathcal{L}_r + d.CFG	0.273	0.749	19.66

Table 2: Exploration of iCD-SD1.5 configurations in terms of image inversion performance.

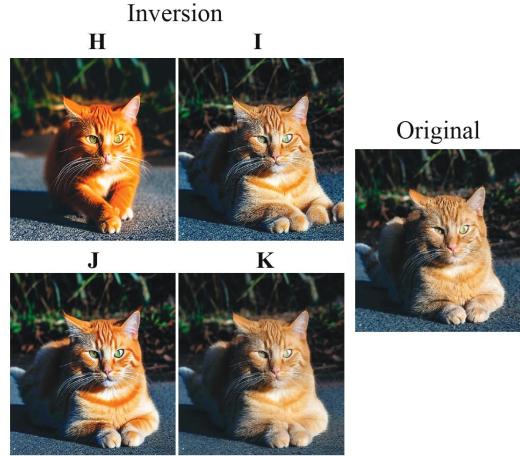


Figure 7: Influence of the dynamic guidance and preservation losses on image inversion with iCD.

CFG ($w = 1$), while for the reverse model, we consider two settings: unguided ($w = 1$) and guided ($w = 8$), both of which are important in practice.

Configuration To evaluate the inversion quality, we consider 5K images and the corresponding prompts from the MS-COCO dataset [56]. We measure the reconstruction quality using LPIPS [58], PSNR and cosine distance in the DinoV2 [59] feature space. As for the reference, the teacher inversion with a disabled CFG scale is considered. For the dynamic guidance, we use $\tau = 0.7$. The coefficients for the preservation losses are equal to $\lambda_f = 1.5$ and $\lambda_r = 1.5$.

Results The results are presented in Table 2. First, configurations (A-C) evaluate the number of the forward and inverse models inference steps. We observe that the reconstruction quality improves as the number of steps increases. In our main experiments, we consider 3 and 4 steps.

Then, (E-G) examine the preservation losses. In (E), we learn the forward model in the encoder [60] regime using the forward preservation loss only. This experiment reveals that the consistency loss contributes significantly to inversion performance. (F, G) show that both losses improve the inversion, with the latter approaching the quality of the teacher model.

Finally, we explore the dynamic CFG and preservation losses under the guided decoding setting (I-K) and compare them to the setting (H), which does not employ any boosting techniques. From the configurations (I, J, K), we can see that all techniques provide significant contribution to the reconstruction quality. In Figure 7, we visualize their influence on inversion. It can be seen that the dynamic CFG (I) is rather responsible for global object preservation, while the preservation losses (J, K) rather improve fine-grained details. We note that the final configuration (K) provides comparable inversion quality to the unguided process while preserving the editing capabilities due to the activated guidance. More visual examples of inversion and quantitative results are in Appendix C.

4.2 Text-guided image editing

In this section, we apply the proposed iCD to the text-guided image editing problem. For the SD1.5 model, we use the Prompt-to-Prompt (P2P) method [61]. We vary two hyperparameters: the cross-attention and self-attention steps balancing between editing strength and preservation of the reference image. We also apply our approach to MasaCTRL [62] in Appendix D. For the SDXL model, we follow the ReNoise [25] evaluation setting and just change the source prompt during decoding according to [63].

Metrics We measure editing performance using both automatic metrics and human-study. The former uses two metrics: 1) to estimate the preservation of the reference image, we calculate the cosine distance between images in the DinoV2 feature space; 2) as an editing quality measure, we use the CLIP score between the edited image and the target prompt. For human evaluation, we employ professional assessors who successfully completed assessment tasks. We show them the source and target prompts, reference image and two images produced with the methods under the comparison

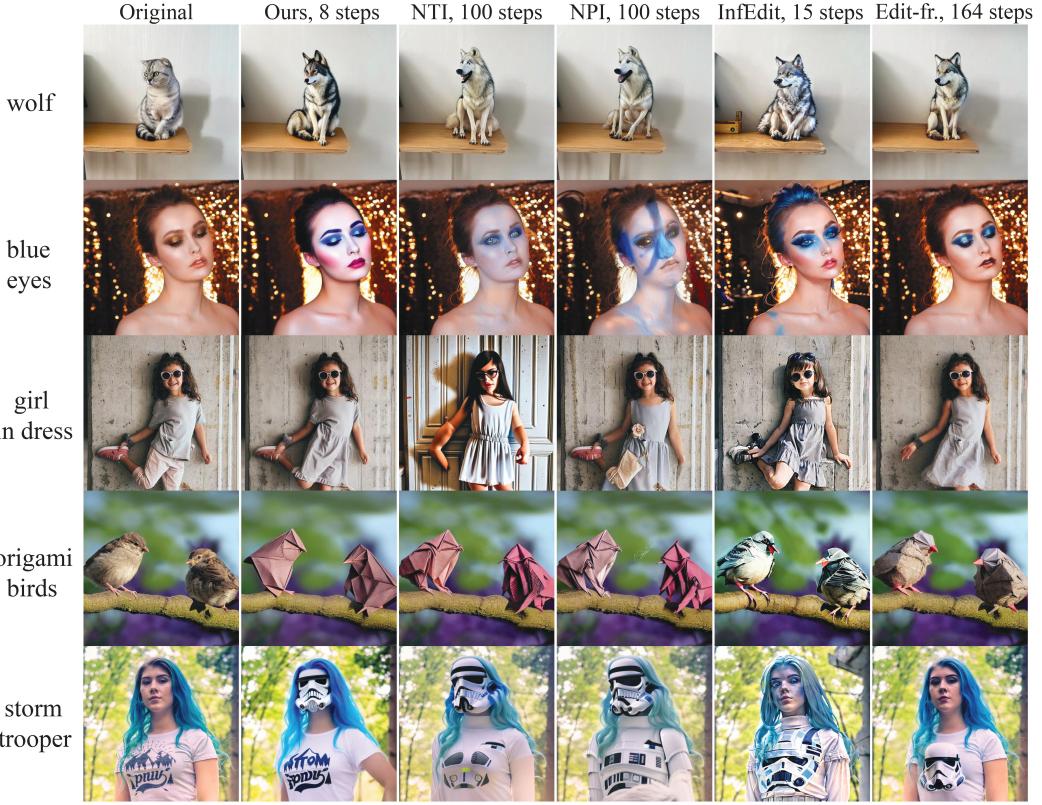


Figure 8: Image editing examples produced by our method (iCD-SD1.5) and the baseline approaches.

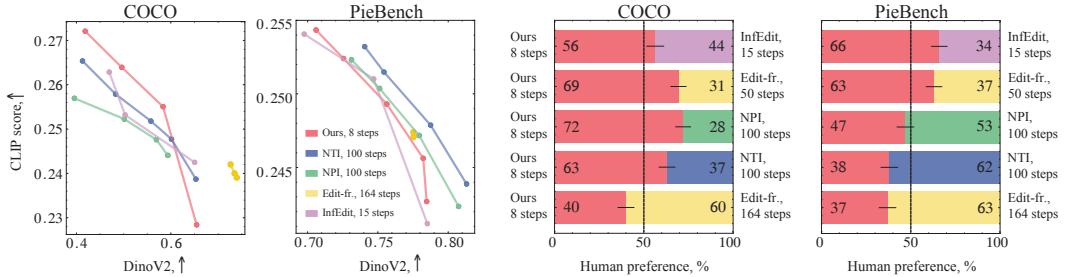


Figure 9: Quantitative comparisons between different editing approaches based on SD1.5: automatic metrics (left) and human preference study (right).

and ask the question: *which of the edited images do you prefer more taking into account the editing strength and reference preservation?*

Benchmarks In our experiments, we consider two benchmarks: PieBench [31] and a manually created COCO evaluation set based on text-paired images from the MS-COCO dataset [56]. PieBench [31] consists of various types of editing, for example, replacement, addition, or deletion. We take 420 examples of realistic images, including all types of editing. COCO focuses solely on the replacement task as one of the most popular among practitioners. This benchmark contains 140 image-text pairs.

4.2.1 Text-guided image editing with iCD-SD1.5

Configuration To provide the editing with the SD1.5 model, we consider iCD using 4 forward and 4 reverse steps trained with both preservation losses ($\lambda_f = 1.5$, $\lambda_r = 1.5$). In Appendix D, we also present the results for iCD using 3 steps, which is not much worse than the 4 step model. We set the hyperparameters of the dynamic CFG to $\tau = 0.8$ and maximum CFG scale to 19.0.

Configuration	Automatic evaluation		
	CLIP score, T \uparrow	DinoV2 \uparrow	CLIP score, I \uparrow
COCO benchmark			
Ours, 8	0.267	0.472	0.748
ReNoise Turbo, 44	0.265	0.399	0.705
ReNoise SDXL, 150	0.227	0.431	0.732
ReNoise LCM, 35	0.218	0.350	0.663
PieBench			
Ours, 8	0.254	0.707	0.858
ReNoise Turbo, 44	0.254	0.615	0.820
ReNoise SDXL, 150	0.234	0.642	0.835
ReNoise LCM, 35	0.236	0.736	0.865
Human preference, %			
COCO benchmark		PieBench	
Ours, 8	64 \pm 4.9	Ours, 8	63 \pm 5.1
ReNoise Turbo, 44	36 \pm 4.9	ReNoise Turbo, 44	37 \pm 5.1
Ours, 8	91 \pm 2.8	Ours, 8	78 \pm 4.3
ReNoise SDXL, 150	9 \pm 2.8	ReNoise SDXL, 150	22 \pm 4.3
Ours, 8	95 \pm 1.8	Ours, 8	76 \pm 4.2
ReNoise LCM, 35	5 \pm 1.8	ReNoise LCM, 35	24 \pm 4.2

Table 3: Automatic metrics (top) and human evaluation (bottom) for iCD-XL and ReNoise [25].



Figure 10: Editing examples using XL models.

Baselines We compare our approach with four baseline approaches, which provide state-of-the-art editing performance: Null-text Inversion (NTI), Negative-prompt Inversion (NPI) [32], InfEdit [26] and Edit-friendly DDPM [43]. All methods except InfEdit are diffusion-based approaches using more than 100 steps. Moreover, NTI employs the additional high-cost optimization procedure to improve inversion quality. InfEdit operates with the distilled diffusion model, latent consistency distillation [18], but utilizing virtual inversion. All methods use P2P and we vary all possible hyperparameter values to find the configurations that provide the best editing-preservation trade-off.

Results Figure 9 provides quantitative results for both benchmarks. We observe that the proposed iCD is comparable to the baseline approaches in most cases. Moreover, sometimes it can even outperform them while being multiple times faster. For instance, according to human preference on the COCO benchmark, our approach surpasses the InfEdit, NPI, NTI and Edit-friendly DDPM (50 steps). On the PieBench, it outperforms the InfEdit and Edit-friendly DDPM (50 steps). In addition, we provide qualitative results in Figure 8, which confirm the competitiveness of the proposed method. Additional visual examples can be found in Appendix D.

Notably, the performance of the proposed method is weaker on the PieBench benchmark compared to COCO, according to both automatic and human-based metrics. We attribute this to the increased complexity of editing tasks, which probably require more steps.

4.2.2 Text-guided image editing with iCD-XL

Configuration We consider the configuration using 4 steps, $\tau = 0.7$ and CFG scale equals 8.0.

Baselines We compare our approach to the recently proposed ReNoise [25], accurately following its guidelines. This method works with both distilled models (LCM-SDXL [18], SDXL-Turbo [19]) and original diffusion model, SDXL [1]. However, even for the distilled models, a significant number of steps is required to achieve decent performance.

Results The quantitative and qualitative comparisons are presented in Table 3 and Figure 10, respectively. According to the human evaluation, iCD-XL outperforms all ReNoise configurations. Based on the automatic evaluation, our approach provides better reference preservation (DinoV2 and CLIP score, I) while maintaining strong editing capabilities, as indicated by the CLIP score (T). We provide more visual examples in Appendix D.

5 Conclusion

The paper proposes a generalized consistency distillation framework that enables both accurate image inversion and solid generation performance using a few inference steps. Accompanied by the recently proposed dynamic guidance, the distilled models demonstrate highly efficient and accurate image manipulations, making a significant step towards real-time text-driven image editing.

References

- [1] 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*, 2024.
- [2] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024.
- [3] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.
- [4] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021.
- [5] 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.
- [6] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- σ : Weak-to-strong training of diffusion transformer for 4k text-to-image generation, 2024.
- [7] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023.
- [8] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. *arXiv preprint arXiv:2311.10089*, 2023.
- [9] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023.
- [10] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.
- [11] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.
- [12] Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. *arXiv preprint arXiv:2310.14189*, 2023.
- [13] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279*, 2023.
- [14] Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [15] David Berthelot, Arnaud Autef, Jierui Lin, Dian Ang Yap, Shuangfei Zhai, Siyuan Hu, Daniel Zheng, Walter Talbott, and Eric Gu. Tract: Denoising diffusion models with transitive closure time-distillation, 2023.
- [16] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

- [17] Liangchen Li and Jiajun He. Bidirectional consistency models. *arXiv preprint arXiv:2403.18035*, 2024.
- [18] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023.
- [19] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation, 2023.
- [20] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.
- [21] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *International Conference on Learning Representations*, 2024.
- [22] Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation, 2024.
- [23] Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation, 2024.
- [24] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédéric Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024.
- [25] Daniel Garabi, Or Patashnik, Andrey Voynov, Hadar Averbuch-Elor, and Daniel Cohen-Or. Renoise: Real image inversion through iterative noising, 2024.
- [26] Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with natural language. 2024.
- [27] Barak Meiri, Dvir Samuel, Nir Darshan, Gal Chechik, Shai Avidan, and Rami Ben-Ari. Fixed-point inversion for text-to-image diffusion models, 2023.
- [28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [29] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.
- [30] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.
- [31] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. *International Conference on Learning Representations (ICLR)*, 2024.
- [32] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023.
- [33] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models, 2024.
- [34] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022.

[35] Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. Inversion-based style transfer with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10146–10156, 2023.

[36] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[37] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. *arXiv preprint arXiv:2211.12446*, 2022.

[38] Senmao Li, Joost van de Weijer, Taihang Hu, Fahad Shahbaz Khan, Qibin Hou, Yaxing Wang, and Jian Yang. Stylediffusion: Prompt-embedding inversion for text-based editing. *arXiv preprint arXiv:2303.15649*, 2023.

[39] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023.

[40] Wenkai Dong, Song Xue, Xiaoyue Duan, and Shumin Han. Prompt tuning inversion for text-driven image editing using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7430–7440, 2023.

[41] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models. *arXiv preprint arXiv:2305.16807*, 2023.

[42] Ligong Han, Song Wen, Qi Chen, Zhixing Zhang, Kunpeng Song, Mengwei Ren, Ruijiang Gao, Yuxiao Chen, Di Liu, Qilong Zhangli, et al. Improving negative-prompt inversion via proximal guidance. *arXiv preprint arXiv:2306.05414*, 2023.

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

[44] Manuel Brack, Felix Friedrich, Katharina Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinarios Passos. Ledits++: Limitless image editing using text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[45] Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann M. Weber. CADS: Unleashing the diversity of diffusion models through condition-annealed sampling. In *The Twelfth International Conference on Learning Representations*, 2024.

[46] Tuomas Kynkänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models, 2024.

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

[48] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

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

[50] 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.

[51] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

[53] Jonathan Heek, Emiel Hoogeboom, and Tim Salimans. Multistep consistency models. *arXiv preprint arXiv:2403.06807*, 2024.

[54] Angela Castillo, Jonas Kohler, Juan C Pérez, Juan Pablo Pérez, Albert Pumarola, Bernard Ghanem, Pablo Arbeláez, and Ali Thabet. Adaptive guidance: Training-free acceleration of conditional diffusion models. *arXiv preprint arXiv:2312.12487*, 2023.

[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.

[56] 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.

[57] 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, June 2022.

[58] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[59] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[60] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: A stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2287–2296, June 2021.

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

[62] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023.

[63] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

[64] 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.

A Training details of iCD

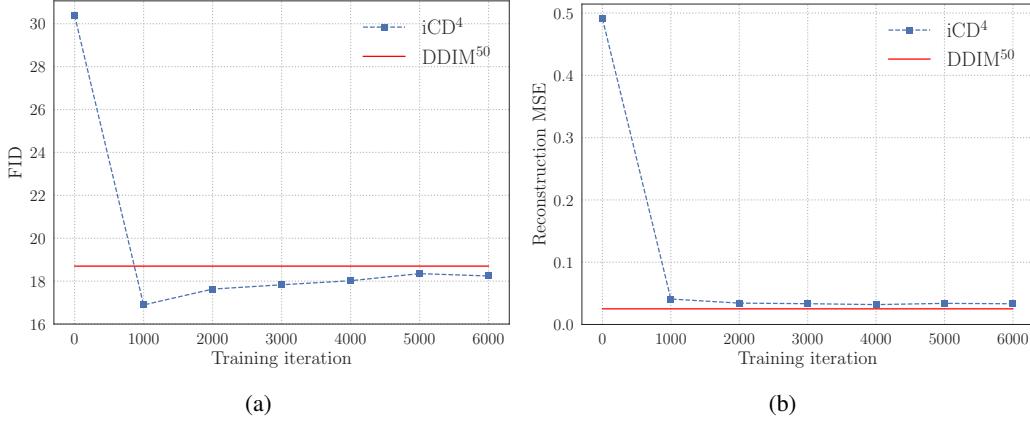


Figure 11: Training dynamics of iCD-SD1.5 in terms of FID (a) and reconstruction loss (b).

First, following [20], we preliminary distill classifier-free guidance using a conditional embedding added to the time step embedding. The goal is to use different w and apply dynamic techniques during inference after consistency distillation. We perform CFG distillation for the following guidance scales: 1, 8, 12, 16, 20 for SD1.5 and 1, 4, 6, 9, 10, 12, 13, 16, 18, 20 for SDXL. At this stage, the model successfully approximates the guided teacher without hurting its performance.

Then, we perform multi-boundary consistency distillation using LoRA adapters with a rank of 64 following [52]. We train forward and reverse adapters in parallel starting from the same teacher: CFG distilled SD1.5 or SDXL. For the SD1.5 model, we use a global batch size of 512, and for the SDXL 128. All models converge relatively fast, requiring about 6K iterations with a learning rate $8e-6$.

We find that the forward and reverse models provide promising generation and inversion quality for 3 or 4 steps. The regularization coefficients for the forward and reverse preservation losses are $\lambda_f=1.5$ and $\lambda_r=1.5$, respectively.

The iCD-SD1.5 models are trained for $\sim 36h$ and the iCD-XL ones for $\sim 68h$ on 8 NVIDIA A100 GPUs. We present the training dynamics in terms of FID and reconstruction MSE for iCD-SD1.5 in Figure 11.

For SD1.5 distillation, we use a $\sim 20M$ subset of LAION 2B, roughly filtered using CLIP score [64]. For SDXL, we collect $\sim 7M$ images with resolution ≥ 1024 , also curated to avoid poorly aligned text-image pairs and low quality images.

We set the following time steps for our configurations:

- 4 steps, $\tau = 0.8$: reverse model [259, 519, 779, 999]; forward model [19, 259, 519, 779];
- 4 steps, $\tau = 0.7$: reverse model [259, 519, 699, 999]; forward model [19, 259, 519, 699];
- 3 steps $\tau = 0.7$: reverse model [339, 699, 999]; forward model [19, 339, 699];

B Image generation with iCD

We provide the generation performance of our distilled model in Table 4. The dynamic CFG ($\tau = 0.8$, $\tau = 0.7$) degrades in terms of ImageReward, while improving FID due to the increased diversity [45, 46]. The visual examples are presented in Figures 16, 17.

C Image inversion

In Table 5, we provide comparisons between the 3- and 4-step configurations of iCD-SD1.5, which perform similarly. Figure 14 shows the image inversions provided by our approach, NTI and NPI.

Configuration	FID	CLIP score	ImageReward
DDIM ⁵⁰	18.73	0.266	0.201
CD ⁴	18.45	0.259	0.044
CD ³	17.96	0.258	-0.009
CD ⁴ + d.CFG, $\tau = 0.8$	16.25	0.254	-0.193
CD ⁴ + d.CFG, $\tau = 0.7$	16.38	0.253	-0.217
CD ³ + d.CFG, $\tau = 0.7$	17.66	0.251	-0.351

Table 4: Text-to-image performance of the SD1.5 model in terms of FID-5K, CLIP score and ImageReward for $w = 8$ using 5K prompts from the MS-COCO dataset.

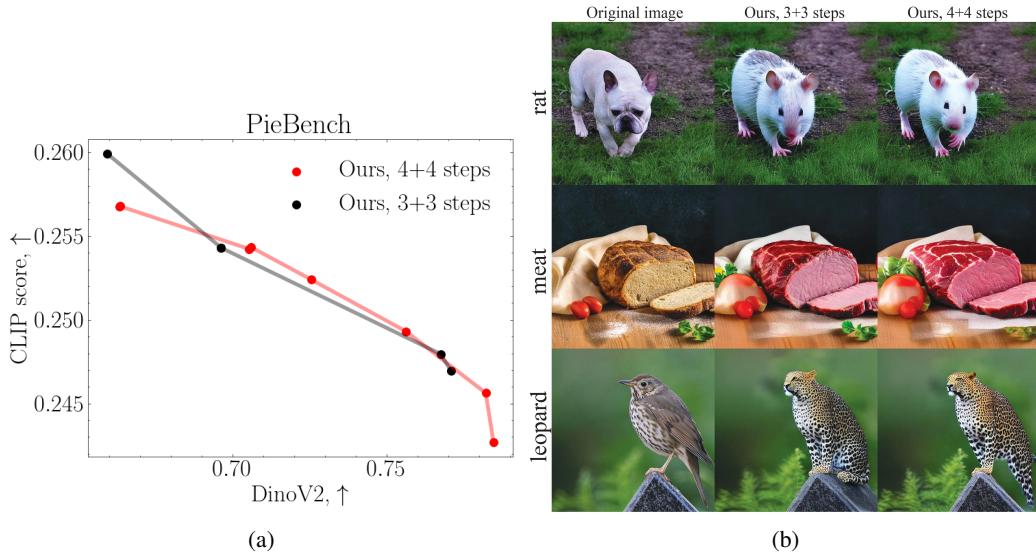


Figure 12: Quantitative (a) and qualitative (b) editing results using 3- and 4-step iCD-SD1.5 configurations on PieBench.



Figure 13: Comparison between Prompt2Prompt and MasaCTRL applied to our approach

Figure 15 shows the image inversions compared to the ReNoise method. In Table 6, we present the time required to invert a single image for different methods.

Configuration	LPIPS ↓	DinoV2 ↑	PSNR ↑
Unguided decoding setting			
fDDIM ⁵⁰ DDIM ⁵⁰	0.167	0.834	22.98
E fCD ⁴ + \mathcal{L}_f CD ⁴ + \mathcal{L}_r	0.198	0.837	22.27
E fCD ³ + \mathcal{L}_f CD ³ + \mathcal{L}_r	0.192	0.841	22.21
Guided decoding setting, $w = 8$			
fDDIM ⁵⁰ DDIM ⁵⁰	0.479	0.534	14.12
fDDIM ⁵⁰ DDIM ⁵⁰ + d.CFG	0.279	0.726	19.58
E fCD ⁴ + \mathcal{L}_f CD ⁴ + \mathcal{L}_r + d.CFG	0.273	0.749	19.66
E fCD ³ + \mathcal{L}_f CD ³ + \mathcal{L}_r + d.CFG	0.263	0.770	19.76

Table 5: More image inversion results for 3- and 4-step iCD-SD1.5. Dynamic CFG uses $\tau = 0.7$.

Method	Ours 8 steps, SD1.5	NTI, SD1.5	NPI, SD1.5	Ours 8 steps, SDXL	ReNoise, LCM-XL
Time, secs	$0.959 \pm .005$	$116.4 \pm .1$	$9.95 \pm .03$	$1.56 \pm .07$	$6.75 \pm .52$

Table 6: The time required to invert a single image for different approaches.

D Text-guided image editing

Figure 12 compares two iCD configurations (3 and 4 steps). We observe that both configurations perform similarly, with a slight preference for the 4-step configuration. We present additional visual results on image editing using the iCD-SD1.5 model in Figure 18 and the iCD-XL model in Figure 19.

Figure 13 provides a few examples of our approach combined with MasaCTRL [62] for non-rigid editing. The results demonstrate that our method can be used with various editing methods.

We also present the comparison with SDXL-Turbo using SDEdit [63] in Figure 20 (upper). We observe that SDXL Turbo significantly hurts reference image preservation due to stochasticity. This highlights the importance of accurate image inversion for editing. Moreover, we compare our approach with Instruct-Pix2Pix [7], Figure 20 (bottom), and observe that it outperforms Instruct-Pix2Pix in terms of both content preservation and editing strength while being training-free.

We present the annotation interface for the assessors in Figure 22. We calculate the confidence interval using bootstrap methods, splitting the human votes into 1,000 subsets and then averaging the results and calculating the standard deviation.

E Failure cases

We present some inversion failure cases in Figure 21. Our method sometimes oversaturates images for high guidance scales and struggles to reconstruct complex details like human faces and hands

F Limitations

The iCD limitations include the requirement to predetermine boundary time steps before distillation, which may restrict the model flexibility. Additionally, the extra preservation losses necessitate increased computational resources during training. Furthermore, the editing method is susceptible to hyperparameter values, leading to inconsistent performance across different prompts. Finally, the quality of the current distillation techniques itself requires significant improvement, as it does not consistently achieve high standards across various scenarios.

G Broader impacts

Our work can significantly enhance tools for artists, designers, and content creators, allowing for more precise and efficient manipulation of images based on textual inputs. This can democratize high-quality digital art creation, making it accessible to those without extensive technical skills. On the other hand, the ability to edit images easily and realistically can be misused to create misleading information or fake images, which can be particularly harmful and potentially influence public opinion and elections.

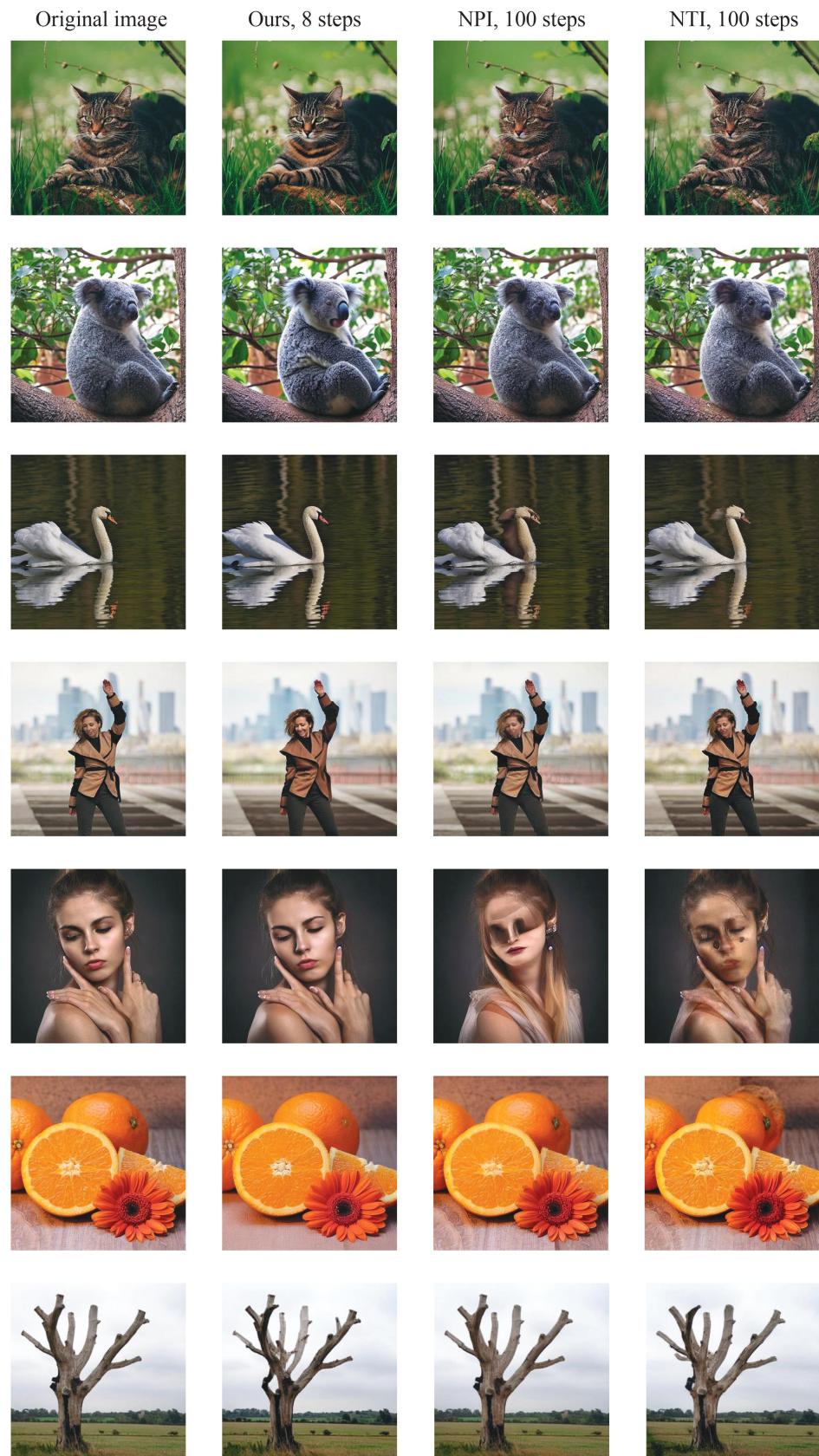


Figure 14: Inversion examples produced by the SD1.5-based models.

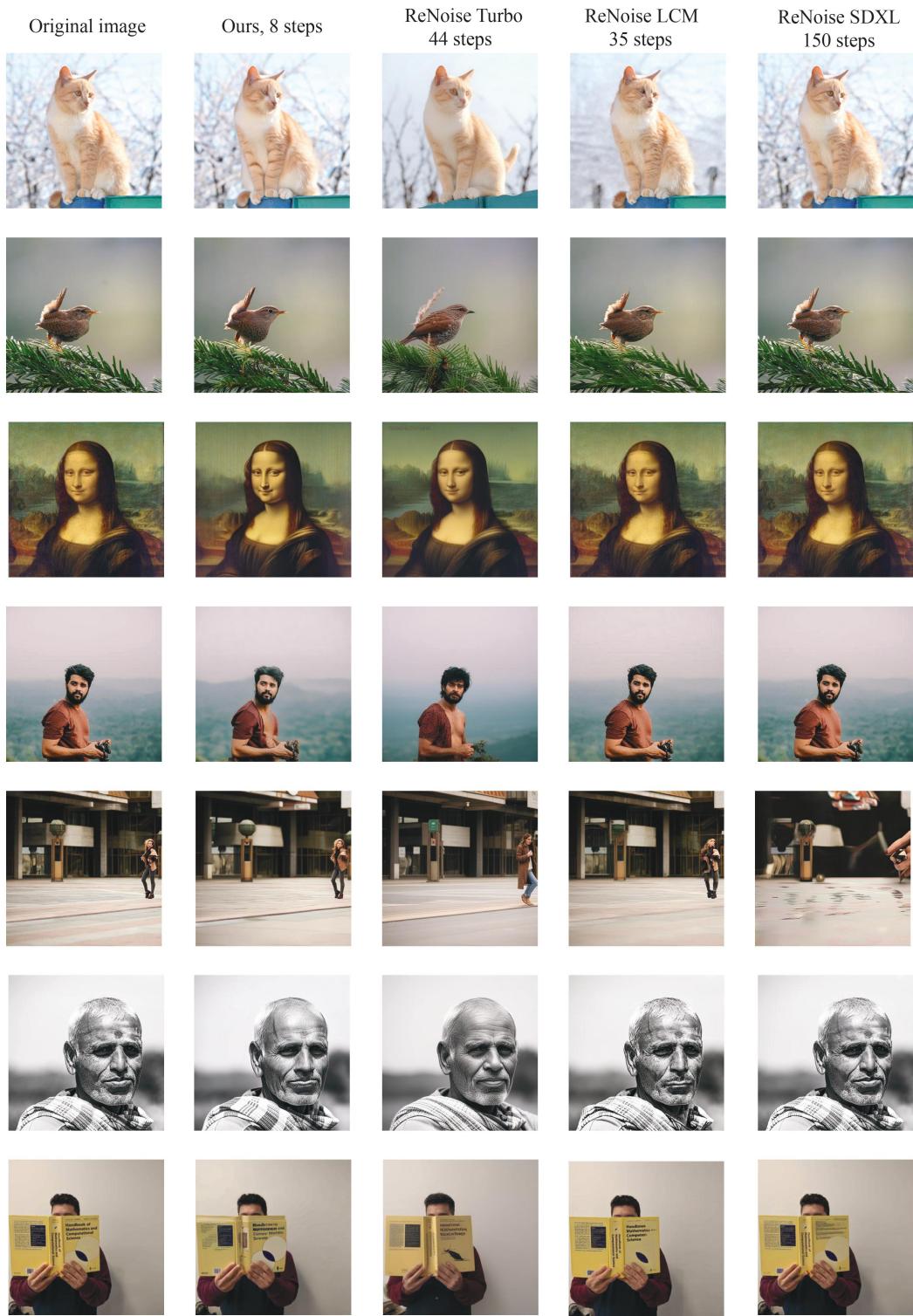
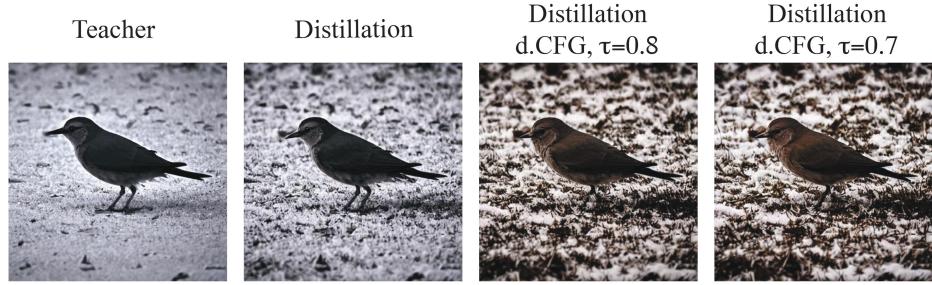


Figure 15: Inversion examples produced by the SDXL-based models.



A grey bird in the middle of a snow-covered ground



A brown stuffed teddy bear sitting on top of a couch.



An elephant at a zoo on a sunny day



Man and woman sitting next to each other



A panoramic view of a cabin with stone fireplace.



A plate with steak, gravy, baked apples, and broccoli.

Figure 16: Generation examples using SD1.5, the proposed distilled method using 4 steps and dynamic CFG



a barred owl peeking out from dense tree branches



a dutch baroque painting of a horse in a field of flowers



a bloody mary cocktail next to a napkin



a cute illustration of a horned owl with a graduation cap and diploma



a peaceful lakeside landscape with migrating herd of sauropods



A blue Ford F-150 coming around a curve in a mountain road

Figure 17: Generation examples using SDXL, the proposed distilled method using 4 steps and dynamic CFG

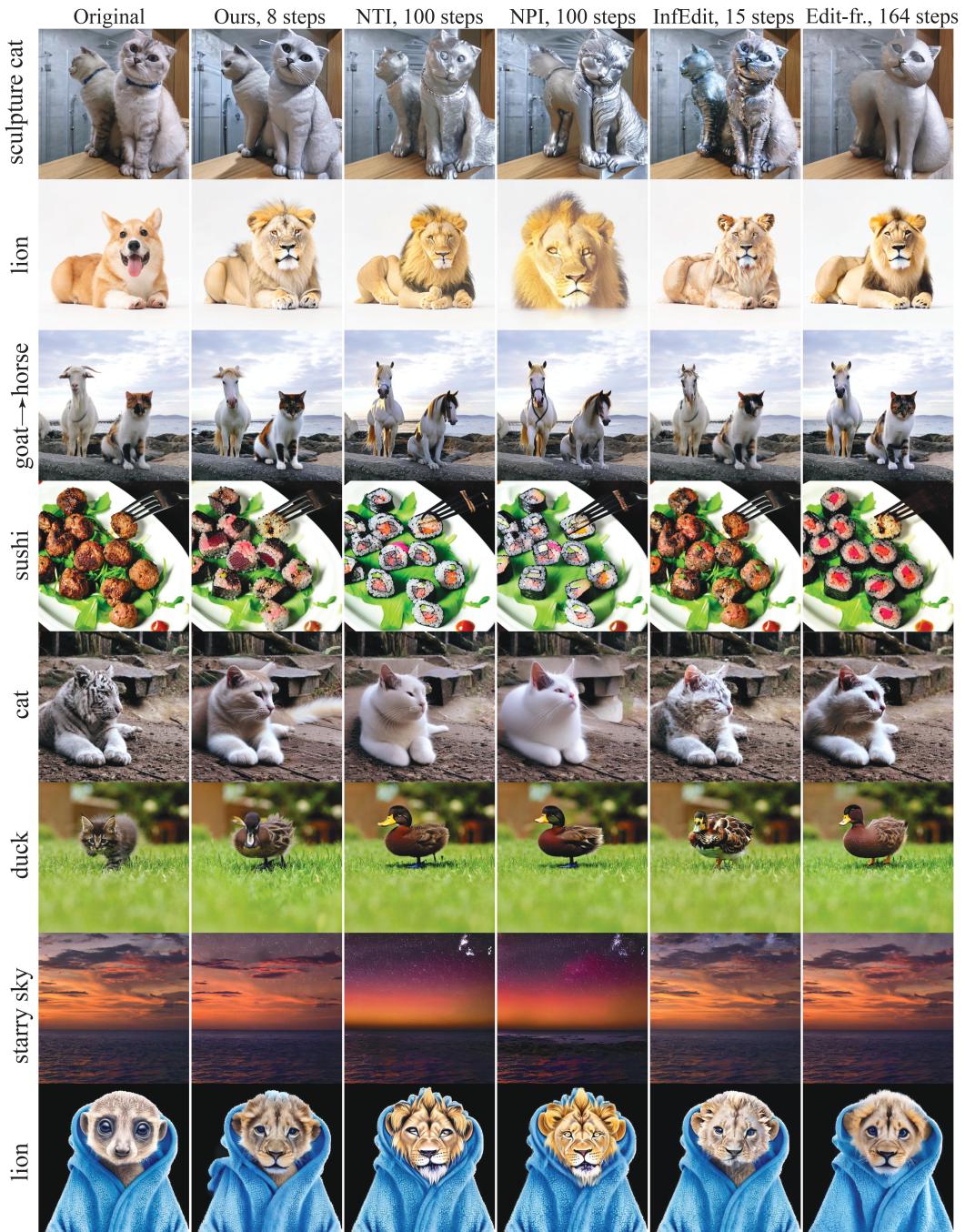


Figure 18: Additional editing examples produced by the SD1.5-based models.

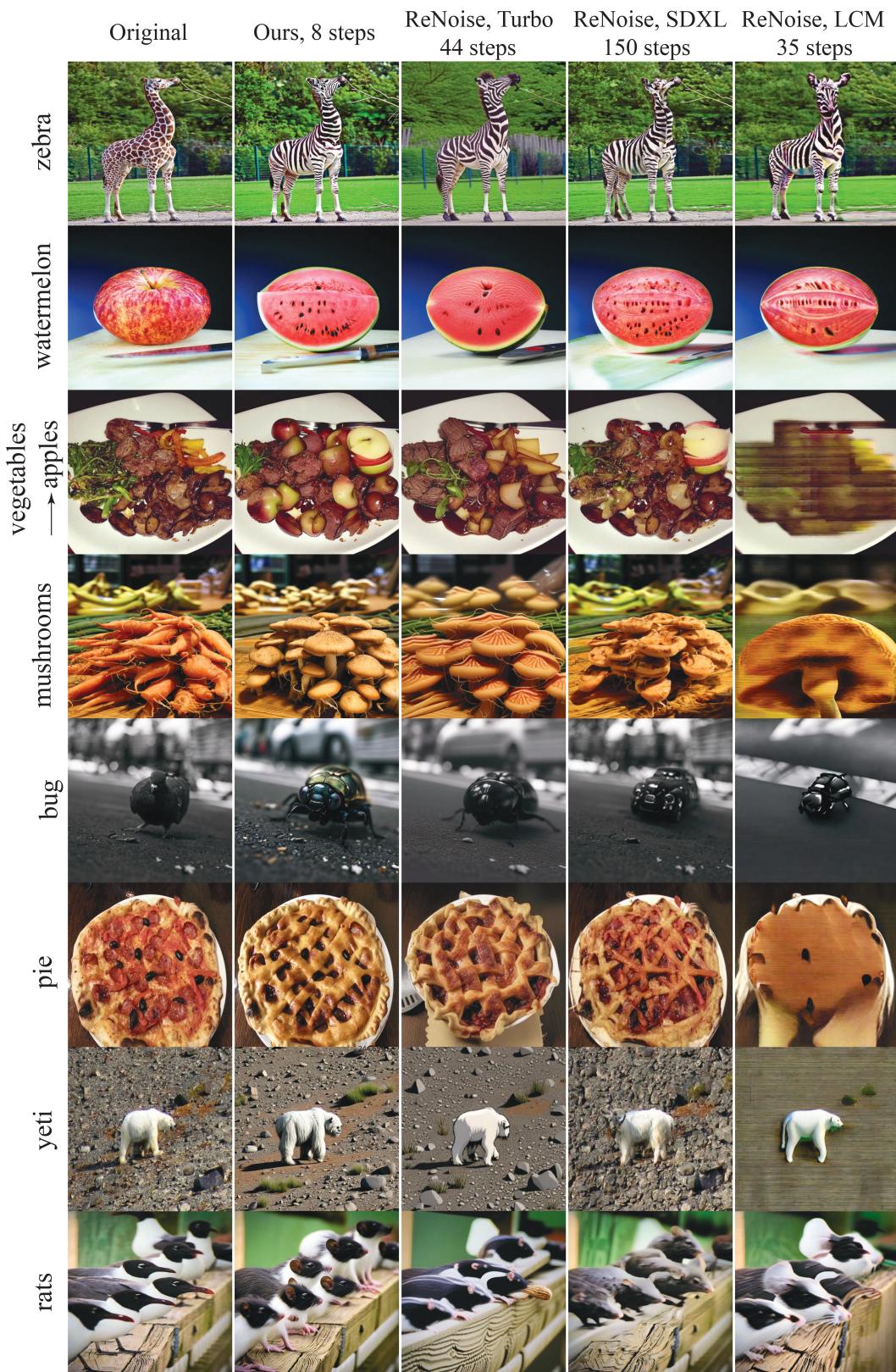


Figure 19: Additional editing examples using the SDXL-based models.

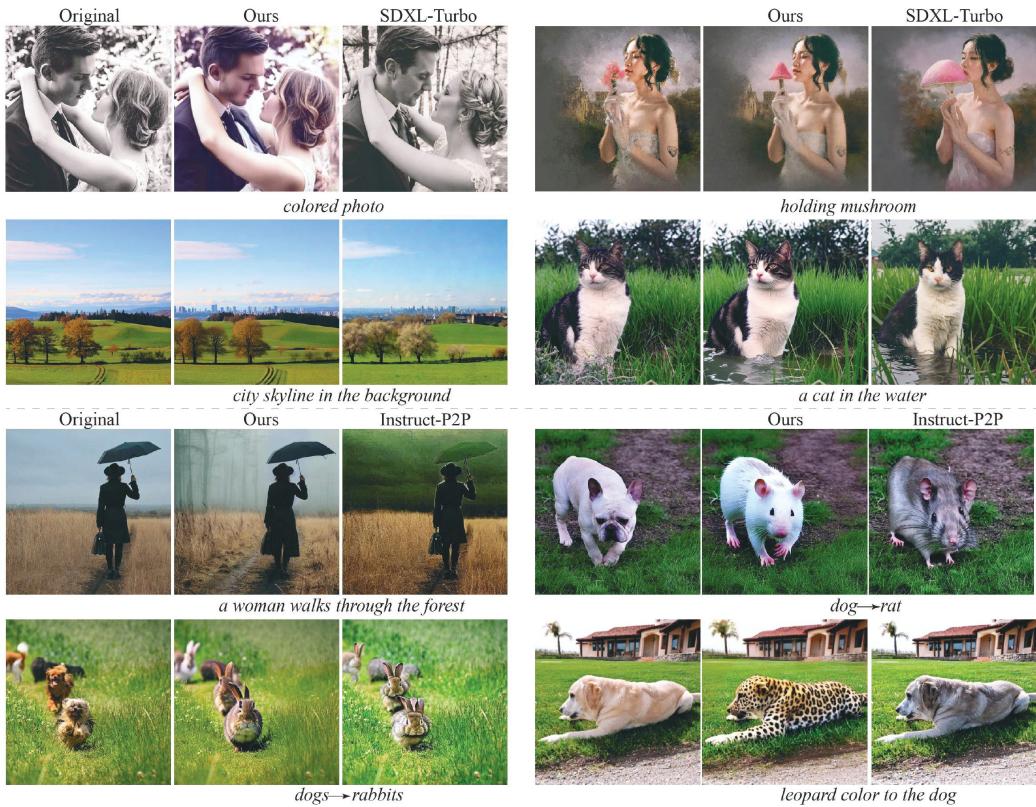


Figure 20: Image editing results produced by our method and SDXL-turbo using SDEdit (upper) and Instruct-P2P (bottom).



Figure 21: Failure cases.

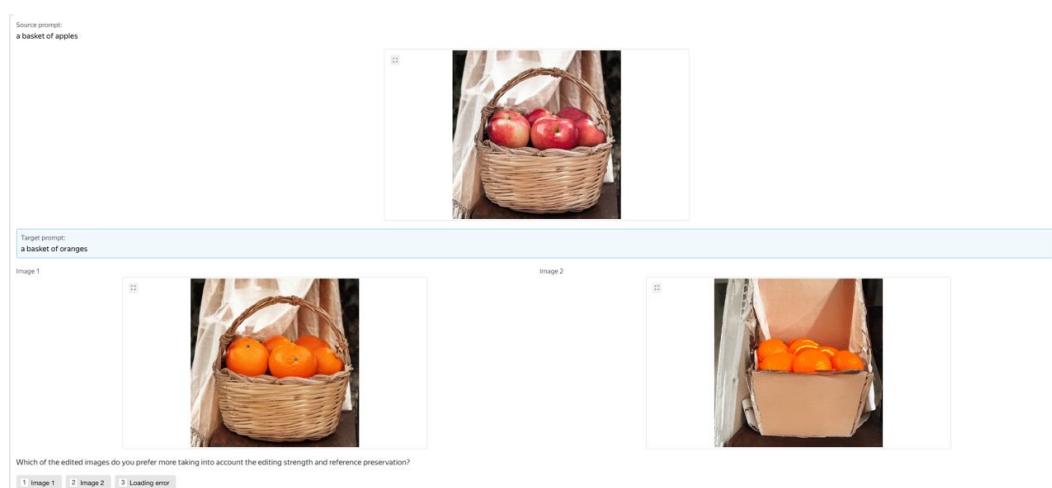


Figure 22: The human evaluation interface for the text-guided image editing problem.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The contributions mentioned in the abstract and introduction sections are carefully aligned with the methods and experimental sections.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: the limitations are discussed in the supplementary material.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#)

Justification: The paper does not provide theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: **[Yes]**

Justification: We provide the technical details and include the code in the supplementary materials.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: We attach the code in the supplementary material and will release it upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: To the best of our judgment, we provide the details sufficient to reproduce the results. However, we do not discuss some low-level implementation details that are not critical for understanding, and refer the readers to our code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: We provide the confidence intervals for the main experiments and explain their nature in the supplementary material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: We provide the configurations in each section of the experiments. Moreover, we include additional information about the training in the supplementary material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [\[Yes\]](#)

Justification: We have carefully read the NeurIPS Code of Ethics and conducted our research in accordance with it.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[Yes\]](#)

Justification: We discuss potential positive societal impacts and negative societal impacts in the supplementary material.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [\[Yes\]](#)

Justification: We will provide guidelines for users upon acceptance of a paper to control the use of our models.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification:

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: We plan to release our code and models upon acceptance under the CC-by 4.0 license.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[Yes\]](#)

Justification: Human evaluation is performed by professional assessors. They are officially hired, paid competitive salary and informed about potential risks. We include the instruction in the supplementary material.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[Yes\]](#)

Justification: Before involving participants in the evaluation, we provide them with detailed instructions addressing potential risks.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.