Applying Guidance in a Limited Interval Improves Sample and Distribution Quality in Diffusion Models

Tuomas Kynkäänniemi
Aalto University

Miika Aittala
NVIDIA

Tero Karras
NVIDIA

Samuli Laine
NVIDIA

Timo Aila
NVIDIA

Jaakko Lehtinen
Aalto University, NVIDIA

Abstract

Guidance is a crucial technique for extracting the best performance out of image-generating diffusion models. Traditionally, a constant guidance weight has been applied throughout the sampling chain of an image. We show that guidance is clearly harmful toward the beginning of the chain (high noise levels), largely unnecessary toward the end (low noise levels), and only beneficial in the middle. We thus restrict it to a specific range of noise levels, improving both the inference speed and result quality. This limited guidance interval improves the record FID in ImageNet-512 significantly, from 1.81 to 1.40. We show that it is quantitatively and qualitatively beneficial across different sampler parameters, network architectures, and datasets, including the large-scale setting of Stable Diffusion XL. We thus suggest exposing the guidance interval as a hyperparameter in all diffusion models that use guidance.

1 Introduction

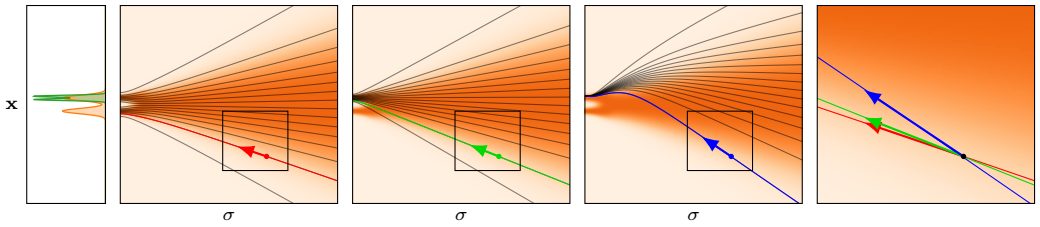
Denoising diffusion models [17, 28, 38, 39, 40, 41, 43, 20] have enabled rapid advances in high-quality image synthesis based on text prompts and other forms of input [13, 35, 44]. They scale effortlessly to large-scale datasets [4, 5, 34], and also to other modalities such as video [8, 7, 16, 19], 3D shapes [26, 31, 33, 37], and audio [24, 32].

Diffusion models convert an initial image of pure noise to a novel generated image through repeated application of image denoising. This sampling chain typically contains dozens of steps, and in each step a little bit of the denoised result is blended into the noisy image. The sampling process first gravitates towards the mean of the training data, followed by the determination of image features in an approximate coarse-to-fine manner based on the remaining noise. This iterative process, where the image is formed little by little, offers considerable flexibility in terms of encouraging or discouraging certain kinds of behavior at each step.

Negative prompts [3] are a widely used concept, where the sampling process is given an additional anti-goal that is to be avoided. For example, "nudity" might be a common negative prompt in text-based image generators. At every sampling step, the denoiser is executed twice: once for the positive and once for the negative prompt, and the positive result is then extrapolated further away from the negative one based on a weight parameter. This works remarkably well in practice. Classifier-free guidance (CFG) [18] builds on this general concept. It uses an unconditional model (no class information or text prompts) as a negative prompt, causing the result image to align more strongly with the conditioning signal.

In practice, all large-scale image generators rely heavily on CFG. It allows a mathematically justified way of truncating the distribution of generated images [12, 18], basically trading variation for

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) PDFs (b) Unconditional (c) Conditional (d) Guided, $w = 6$ (e) Tangent vectors

Figure 1: Visualizing the effect of guidance. **(a)** The unconditional (orange) and conditional (green) PDFs. In (b) through (e), the orange unconditional density is visualized in the background. **(b)** Sample trajectories for the unconditional distribution. **(c)** Trajectories for the conditional distribution. **(d)** Trajectories for the guided distribution with $w = 6$. **(e)** The tangent vectors $\mathrm{d}\mathbf{x}/\mathrm{d}\sigma$ at the intersection point of the three marked trajectories. The difference of the unconditional (red) and conditional (green) vectors is magnified as per Equation 4, causing the unexpected detour in low-probability areas and a mode drop. See Figure 2 for details and comparison to our approach.

perceptually higher image quality. By convention, the same guidance weight is used in all sampling steps. We observe that this is sub-optimal because CFG behaves very differently on high, middle, and low noise levels. On high noise levels, it drastically reduces the variation in the results, basically leading them towards a handful of "template images" per prompt. On middle levels, it causes the sampling to more decisively choose some set of features, leading to crisper and perceptually more pleasing results. On low levels, it is largely unnecessary. Similar observations have been made in the Stable Diffusion community [1, 2, 21], and Muse [10] and Masked DiT v2 [14] observe that making guidance weight noise level-dependent improves the results. In the context of prompt inversion, Mahajan et al. [27] notice that limiting the inversion to specific noise levels leads to improved result quality. However, these works do not quantify the effect on distribution metrics with the exception of Sadat et al. [36], whose "dynamic CFG" limits a linearly varying guidance weight to an interval of noise levels. Interestingly, they conclude that dynamic CFG leads to rather poor results, while a more complicated condition annealing scheme is required for good quantitative results.

We suggest that guidance should be simply limited to an interval of sampling steps in the middle, where the net effect is positive, without otherwise changing the guidance weight. This avoids most of the detrimental effects of guidance, while also reducing computational cost. We show that an optimal guidance interval improves the state-of-the-art FID [23] in ImageNet-512 from 1.81 to 1.40 and also leads to an improved visual quality. The benefits are consistent across sampler parameters, network architectures, and datasets, including Stable Diffusion XL. Code is available at `https://github.com/kynkaat/guidance-interval`

2 Background

The concepts in this and the following section are illustrated in Figure 1 using a synthetic 1D example. In this example, generation is performed by ideal analytic denoisers, avoiding all approximations that a learned denoiser might cause. While this renders classifier-free guidance strictly harmful in the scenario, the example allows us to intuitively visualize the kinds of harm it causes.

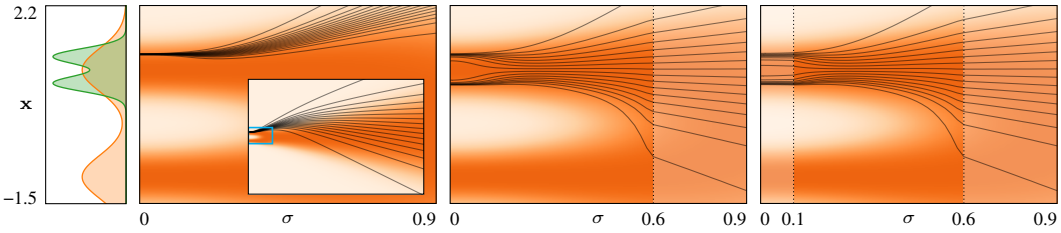
The goal of a denoising diffusion model is to draw samples from a data distribution $p_{\text{data}}(\mathbf{x})$. Let us define a series of smoothed distributions $p(\mathbf{x}; \sigma)$, so that each individual distribution is the convolution between p_{data} and a Gaussian noise distribution with standard deviation σ. Following the EDM formulation [22], the evolution of a sample $\mathbf{x} \sim p(\mathbf{x}; \sigma)$ w.r.t. a change in σ is described by the ordinary differential equation (ODE):

$$\mathrm{d}\mathbf{x}/\mathrm{d}\sigma = -\big(D_\theta(\mathbf{x}; \sigma) - \mathbf{x}\big)/\sigma, \tag{1}$$

where D_θ is a denoiser model with parameters θ, optimized to minimize the expected L_2 denoising error:

$$\theta = \operatorname{argmin}_\theta \, \mathbb{E}_{\mathbf{y} \sim p_{\text{data}}, \sigma \sim p_{\text{train}}, \mathbf{n} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})} \| D_\theta(\mathbf{y} + \mathbf{n}; \sigma) - \mathbf{y} \|_2^2. \tag{2}$$

Here, $p_{\text{train}}(\sigma)$ is the training distribution of noise levels, which we consider to be an implementation detail of D_θ. To generate a sample from the data distribution, we first draw an initial sample



(a) PDFs (b) Guidance everywhere (c) Guidance at $\sigma \in (0, 0.6]$ (d) Guidance at $\sigma \in (0.1, 0.6]$

Figure 2: Illustration of the detrimental effects of guidance at high σ in a synthetic 1D scenario. **(a)** PDFs of the unconditional (orange) and conditional (green) data distributions used in this example. **(b)** Activating guidance (weight $w = 6$) everywhere leads to a catastrophic mode drop. The zoomed-out inset shows how guidance pushes the sampling trajectories outside the distribution during early sampling. **(c)** Disabling guidance at high σ resolves the issue and restores both modes. **(d)** Disabling guidance at low σ has little effect and can be done to reduce computational cost.

$\mathbf{x}_0 \sim p(\mathbf{x}; \sigma_{\text{max}})$, where σ_{max} is chosen to be large enough so that $p(\mathbf{x}; \sigma_{\text{max}})$ is approximately equal to pure Gaussian distribution and thus trivial to sample from. We then follow the ODE of Equation 1 to evolve $\mathbf{x}_0$ towards $\sigma = 0$, i.e., the data distribution. Figure 1a illustrates the target distribution (orange). Figure 1b depicts the diffused target distribution over the σ axis and a set of sample trajectories computed by solving Equation 1 from several different initial conditions.

We can think of classifier-free guidance [18] as constructing a modified ODE where $\mathrm{d}\mathbf{x}/\mathrm{d}\sigma$ is defined as a linear combination between a conditional ODE and an unconditional ODE:

$$\mathrm{d}\mathbf{x}/\mathrm{d}\sigma = w\big[-\big(D_\theta(\mathbf{x}|\mathbf{c};\sigma) - \mathbf{x}\big)/\sigma\big] + (1-w)\big[-\big(D_\theta(\mathbf{x};\sigma) - \mathbf{x}\big)/\sigma\big] \tag{3}$$

$$= -\big(wD_\theta(\mathbf{x}|\mathbf{c};\sigma) + (1-w)D_\theta(\mathbf{x};\sigma) - \mathbf{x}\big)/\sigma, \tag{4}$$

where w is the guidance weight and $\mathbf{c}$ is the condition information given to the denoiser D_θ (cf. Figure 1c). Setting $w > 1$ results in *extrapolating* the effect of the condition with respect to the unconditional result, i.e., the sample is effectively pushed away from the unconditional result. This extrapolation can be seen [12, 18] as raising the conditional likelihood $p(\mathbf{c}|\mathbf{x};\sigma)$ to a power greater than one, which, intuitively, aims to concentrate the probability mass to the regions that most agree with the condition. However, as illustrated in Figure 1(d, e) and the next section, this "oversteering" may direct the trajectories away from the data distribution and cause mode drops.

Most commonly, a single denoiser model D_θ is trained to accept either conditional or unconditional input by dropping the conditioning information 10–20% of the time during training. Alternatively, we can train two separate models $D_{\theta_{\text{cond.}}}(\mathbf{x}|\mathbf{c};\sigma)$ and $D_{\theta_{\text{uncond.}}}(\mathbf{x};\sigma)$. This makes it possible to reduce the capacity of the unconditional model considerably to improve the overall sampling speed [23].

Sampling the ODE is done by taking a number of discrete steps that bring the noise level from σ_{max} to zero, giving rise to a sequence of images $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_N$, each with its corresponding noise level σ_i. Various discretization schemes and solvers have been proposed [22]. Regardless of the specifics, the computational cost is directly proportional to the number of sampling steps N.

3 Our method

In Figure 2, we continue to probe the downsides of CFG using the previous toy example. We observe that applying guidance at all noise levels — as is typical — causes the sampling trajectories to drift quite far from the the smoothed data distribution (Figure 2b). This is caused by the unconditional trajectories effectively repelling the guided trajectories, as discussed above, yielding badly skewed intermediate distributions. As a result, the sampler drops one of the modes (almost) entirely.

As most of the drift seems to be caused at high noise levels, we disable CFG in those sampling steps (Figure 2c). This correctly recovers both modes of the conditional distribution. In addition, disabling guidance at low noise levels (Figure 2d) has only a small effect on the resulting distribution, providing a simple way to reduce the sampling cost with minimal effect on outputs.

Although this toy example is grossly simplified, we hypothesize that broadly similar effects occur in full-scale diffusion models as well. In Section 4 we can see, e.g., image compositions becoming

ImageNet-512		Quality metric		Model size		Guidance interval		Guidance weight	
		FID ↓	FD_{DINOv2} ↓	Mparams	Gflops	FID	FD_{DINOv2}	FID	FD_{DINOv2}
EDM2-S	[23] w/ CFG [18]	2.23	52.32	280	102	Full	Full	1.4	1.9
EDM2-S	[23] w/ guidance interval	1.68	46.25	280	102	$(0.28, 2.90]$	$(0.60, 5.00]$	2.1	3.2
EDM2-XXL	[23] w/ CFG [18]	1.81	33.09	1523	552	Full	Full	1.2	1.7
EDM2-XXL	[23] w/ guidance interval	**1.40**	**29.16**	1523	552	$(0.19, 1.61]$	$(0.60, 5.00]$	2.0	2.9
DiT-XL/2	[29] w/ CFG [18]	3.04	51.97	675	525	Full	Full	1.5	2.0
DiT-XL/2	[29] w/ guidance interval	2.40	43.94	675	525	$(0.34, 1.02]$	$(0.45, 1.23]$	2.5	4.0

Table 1: Quantitative results on ImageNet-512. Limiting the classifier-free guidance (CFG) to an interval improves both FID and FD_{DINOv2} significantly, without altering the model complexity. The sampling cost is a bit lower due to fewer guidance evaluations. This holds for a small (S) and large (XXL) variants of the state-of-the-art EDM2 model [23], as well as diffusion transformers [29]. The model complexity numbers are copied from the EDM2 paper.

less varied due to guidance, somewhat akin to the mode dropping observed in the toy example. That behaviour is difficult to explain by local sharpening of probability distributions alone (Section 2).

3.1 Practice

Motivated by the above observations, we propose to only apply guidance in a continuous interval of noise levels in the middle of the sampling chain and disable it elsewhere. Concretely, we redefine the ODE of Equation 4 by replacing w with a piecewise constant function:

$$\mathrm{d}\mathbf{x}/\mathrm{d}\sigma \;=\; -\Big(w(\sigma)D_\theta(\mathbf{x}|\mathbf{c};\sigma) + \big(1-w(\sigma)\big)D_\theta(\mathbf{x};\sigma) - \mathbf{x}\Big)/\sigma, \tag{5}$$

$$\text{where}\quad w(\sigma) \;=\; \begin{cases} w & \text{if } \sigma \in (\sigma_{\text{lo}}, \sigma_{\text{hi}}] \\ 1 & \text{otherwise.} \end{cases} \tag{6}$$

Here, σ_{hi} denotes the point in the sampling chain where we enable guidance and σ_{lo} is the point where we turn it off. In our formulation, traditional CFG is recovered by setting $\sigma_{\text{lo}} = 0$ and $\sigma_{\text{hi}} = \infty$.

Virtually all existing deterministic samplers can be seen as numerical Runge–Kutta solutions to the ODE of Equation 4, obtained through a number of discrete steps. While the correspondence might not be obvious in all cases, we can nevertheless characterize the steps with respect to σ as detailed in Appendix A. For example, in the case of Stable Diffusion XL [30], we have 32 steps corresponding to the transitions $\sigma_0 \to \sigma_1,\ \sigma_1 \to \sigma_2, \ldots,\ \sigma_{31} \to \sigma_{32}$, where $\sigma_0 = 14.61,\ \sigma_1 = 13.41,\ \sigma_2 = 12.28, \ldots,\ \sigma_{31} = 0.03$, and $\sigma_{32} = 0$.

The underlying assumption common to all Runge–Kutta methods is that $\mathrm{d}\mathbf{x}/\mathrm{d}\sigma$ should be sufficiently smooth within each step. In Equation 6, however, we intentionally introduce discontinuities at σ_{lo} and σ_{hi}. In order to satisfy the smoothness requirement, we must thus ensure that both transitions happen exactly at step boundaries so that the value of $w(\sigma)$ stays constant within each step. In practice, we choose to do this by rounding σ_{lo} and σ_{hi} appropriately, i.e., by setting $\sigma_{\text{hi}} = \sigma_i$ and $\sigma_{\text{lo}} = \sigma_j$ for some $i < j$. Note that this leads to a seemingly high numerical precision in the values of σ_{lo} and σ_{hi}, which should not be taken as an indication of extremely precise tuning.

4 Results

We will first evaluate and ablate our method quantitatively using ImageNet [11]. Limiting the guidance interval leads to clearly identifiable qualitative changes in the images, which we subsequently demonstrate also in the large-scale context using Stable Diffusion XL [30]. Please refer to Appendix B for additional results.

4.1 Main results

We mainly evaluate our method on ImageNet at 512×512, using the current state-of-the-art approach EDM2 [23] as a baseline.[1] We use the small (EDM2-S) and the largest (EDM2-XXL) models as-is with the default sampling parameters: 32 deterministic steps with a 2^{nd} order Heun sampler [22].

[1] `https://github.com/NVlabs/edm2`

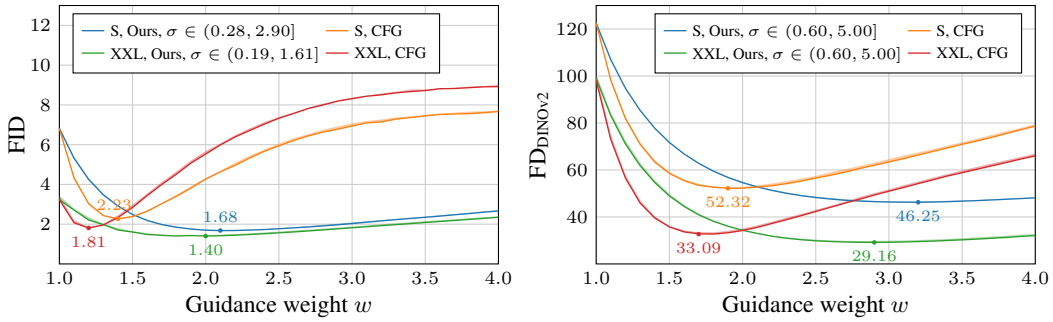


Figure 3: FID and $\text{FD}_{\text{DINOv2}}$ as a function of guidance weight for classifier-free guidance (orange, red) and our method where the guidance has been limited to the stated interval (blue, green). The shaded regions indicate the min/max over three evaluations.

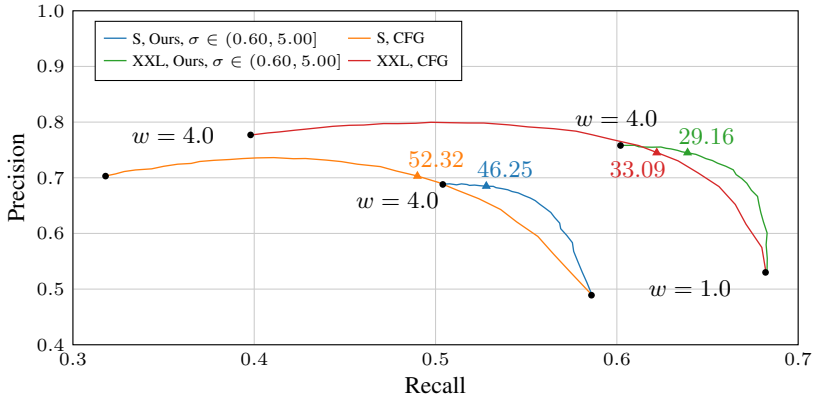


Figure 4: Precision and recall curves for classifier-free guidance (orange, red) and our method (blue, green), when the guidance weight w is varied from 1.0 to 4.0 in 0.1 increments. Black points indicate the minimum and maximum guidance weights in the sweep, while colored triangles show the precision/recall tradeoffs that achieve the best $\text{FD}_{\text{DINOv2}}$. We used the DINOv2 feature space in this plot, following the recommendation by Stein et al. [42]. The curves represent median over three evaluations.

Table 1 shows that our method improves FID [15] and the more recently proposed $\text{FD}_{\text{DINOv2}}$ [42] significantly. Using EDM2-S, FID improves from 2.23 to 1.68, already beating the state-of-the-art in this dataset. With EDM2-XXL, the record further improves to 1.40 and $\text{FD}_{\text{DINOv2}}$ also improves from 33.09 to 29.16.

To find the optimal parameters for each case, we performed a full grid search over w, σ_{lo}, and σ_{hi}. In the case of EDM2-XXL, the best FID is achieved by applying guidance at 6 of the 32 steps, corresponding to noise levels $\sigma \in (0.19, 1.61]$, with weight $w = 2.0$. The best $\text{FD}_{\text{DINOv2}}$ is obtained with slightly higher noise levels $\sigma \in (0.60, 5.00]$ and a slightly higher weight $w = 2.9$.

For additional validation, we also tested our method on diffusion transformers [29] using the DiT-XL/2 model[2] with default sampling parameters: 250 step iDDPM [28]. Limiting the guidance interval leads to significant improvements with this model as well. The best FID results were obtained by using guidance with $w = 2.5$ in 75 of the 250 sampling steps, corresponding to the interval $\sigma \in (0.34, 1.02]$. The best $\text{FD}_{\text{DINOv2}}$ is again obtained with slightly higher noise levels $(0.45, 1.23]$ and weight $w = 4.0$.

4.2 Ablations

Figure 3 shows that standard classifier-free guidance is quite sensitive to the guidance weight. When the weight is too high, the output image distribution is excessively truncated, and the harm caused outside the useful interval starts to outweigh the benefits obtained within. In contrast, limiting the

[2] `https://github.com/facebookresearch/DiT`

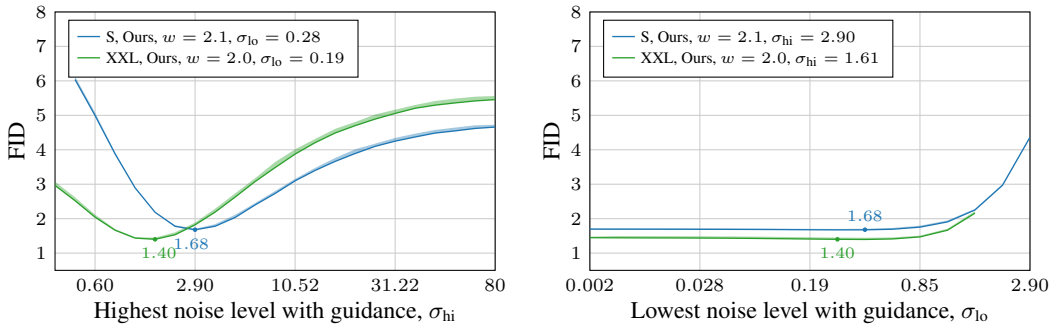


Figure 5: Sensitivity of FID to the chosen guidance interval. **Left:** Sweep over σ_{hi} with optimal σ_{lo} and w. **Right:** Sweep over σ_{lo} with optimal σ_{hi} and w. The shaded regions indicate the min/max over three evaluations.

guidance interval allows the use of much higher guidance weight, and FID or $\text{FD}_{\text{DINOv2}}$ are far less sensitive to the exact choice.

Figure 4 shows precision and recall [25] curves for CFG and our method, evaluated with varying guidance weights in DINOv2 feature space, as suggested by Stein at al. [42]. Compared to CFG, our method achieves better $\text{FD}_{\text{DINOv2}}$ primarily by improving Recall without significantly affecting Precision. This is consistent with the qualitative observation that the results are more varied.

Figure 5 probes the sensitivity of our results to the chosen guidance interval. In this test, we sweep over σ_{lo} and σ_{hi}, while keeping the other interval endpoint, σ_{hi} or σ_{lo}, and the guidance weight w as the optimal choices as reported in Table 1. The left side shows a sweep over σ_{hi}, i.e., the highest noise level with guidance. Including too high noise levels to the guidance interval leads to truncation of the image distribution, which can be seen as an increase in FID. Furthermore, too narrow an interval (low σ_{hi}) yields sub-optimal results. For both EDM2 models the optimal choice for σ_{hi} is located at the middle noise levels. The right side shows a sweep over σ_{lo}, i.e., the lowest noise level with guidance. Applying guidance at low noise levels does not bring additional benefits, compared to the middle levels. Thus, guidance can be disabled in most of the low noise levels to decrease sampling cost, an observation also made in [9].

To estimate the optimal guidance interval in practice, the upper and lower guidance limits can be determined separately, without the need for a two-dimensional search. This happens by first establishing the optimal upper limit by keeping the lower limit at zero. This can be done because the lower limit affects the result only weakly, and in a predictable way (Figure 5, right). Once the optimal upper limit is known, the lower limit is determined. Optionally, a bisection method can be used for accelerating both search operations. Finally, one can reduce the sample size of FID evaluation from 50k to, say, 5k, at least for an initial run, which accelerates the process by $10\times$.

We have found that the optimal choice of σ_{lo} and σ_{hi} is not overly sensitive to the other sampling parameters. For example, if we halve or double the number of steps with EDM2-S, the optimal guidance interval remains unchanged. With 16 steps, our method improves FID from 2.49 to 1.84, and with 64 steps, from 2.27 to 1.70.

In an additional test, we tried applying various smooth weighting functions to the guidance weight (less guidance towards the ends of the interval), but these tests did not improve the results over the simple binary inclusion. We also tried estimating the importance of guidance at individual noise levels by enabling or disabling it at each sampling step at a time. However, these tests consistently underestimated the downsides of guidance, suggesting that they build up cumulatively over multiple consecutive steps.

4.3 Qualitative analysis

With the rise of recent large-scale image generators, ImageNet can hardly be considered a meaningful benchmark for gauging perceptual image quality. Thus, we primarily focus on evaluating our method in the context of Stable Diffusion XL (SD-XL), but we also provide corresponding results for

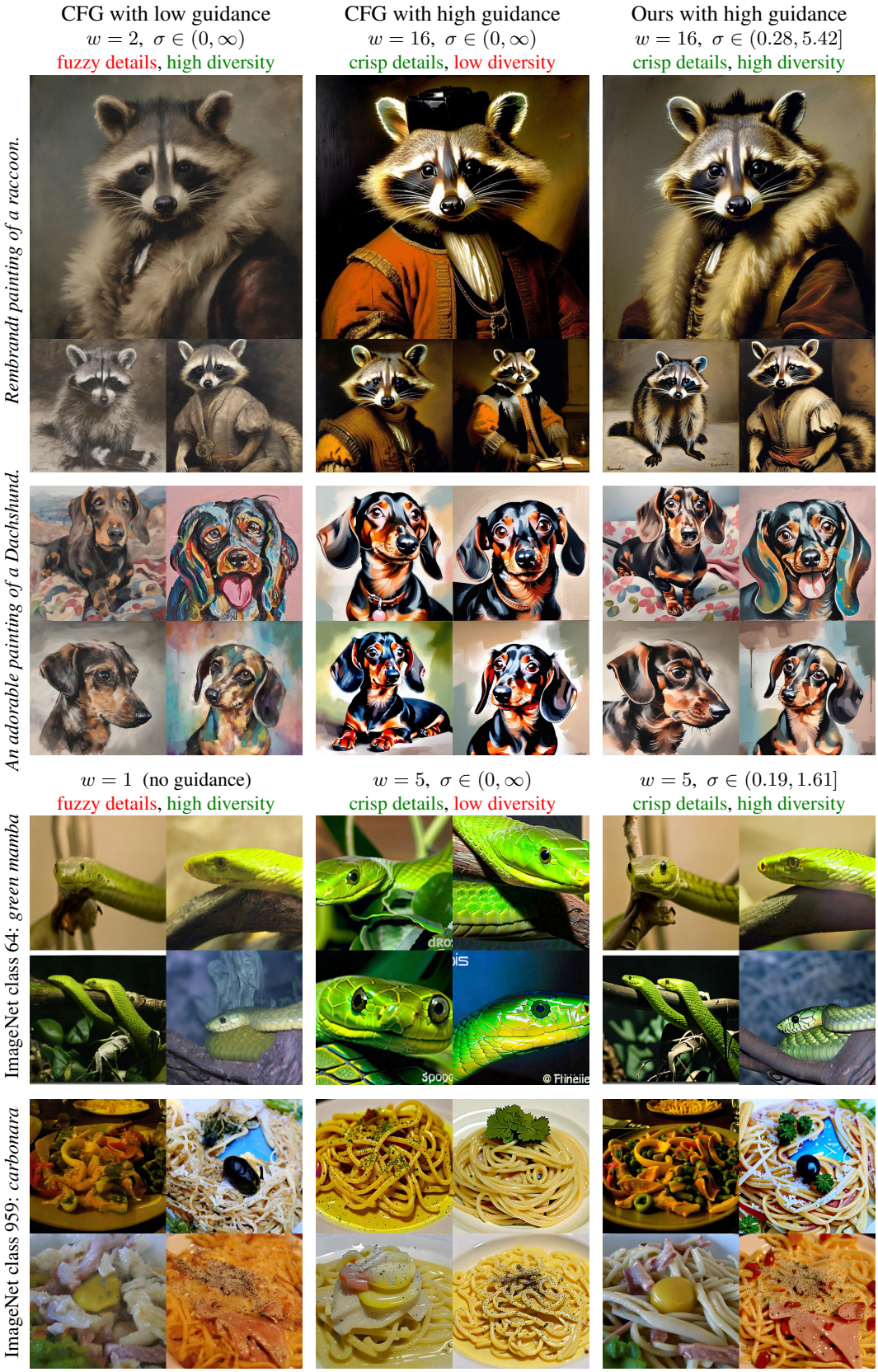
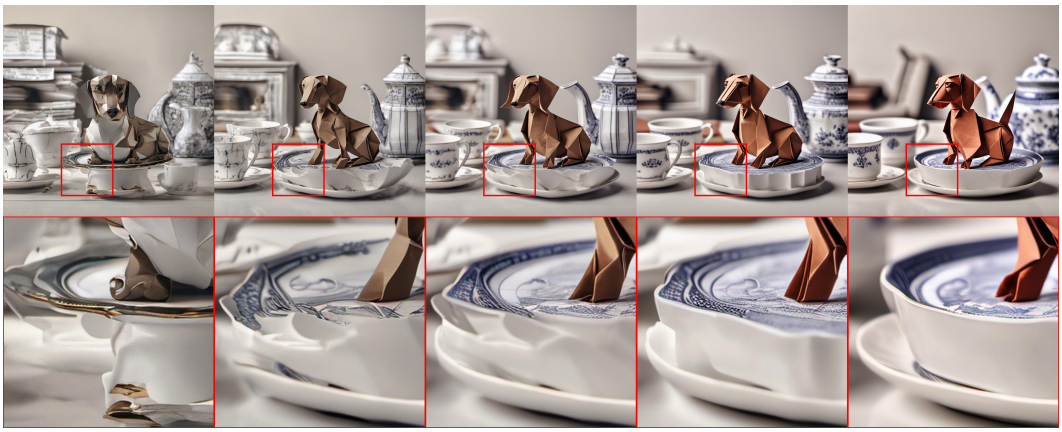
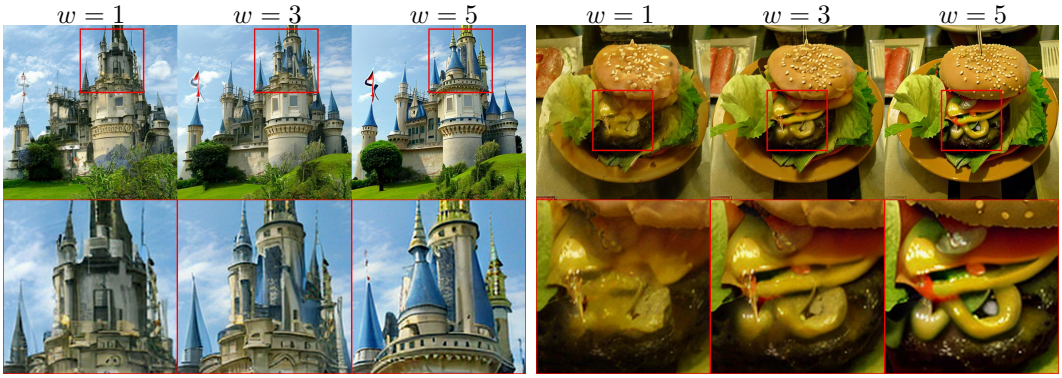


Figure 6: Traditional CFG vs. our method. **Left:** Low w yields diverse but fuzzy images that lack detail. **Middle:** Increasing w adds crispness, but reduces diversity and oversaturates the colors. **Right:** Our method reduces these effects while retaining the crisp look.

A 4k dslr photo of a raccoon wearing an astronaut helmet, photorealistic.



A highly detailed paper origami of a Dachshund on a table next to a porcelain teapot, 4k dslr.



ImageNet class 483: *castle* ImageNet class 933: *cheeseburger*

Figure 7: Effect of guidance weight w with our method. We limit the guidance to $\sigma \in (0.28, 5.42]$ with SD-XL (top) and to $\sigma \in (0.19, 1.61]$ with EDM2-XXL (bottom). Higher w leads to clearer and more well-defined image details while keeping the color palette and overall composition unchanged.

ImageNet using EDM2-XXL. For SD-XL, we use the official pre-trained checkpoint[3] with a standard 32-step deterministic Heun sampler, where the first step corresponds to $\sigma = 14.61$.

With SD-XL, we apply guidance at 50% of the sampling steps, corresponding to noise levels $\sigma \in (0.28, 5.42]$, with weight $w = 16$. These parameters were chosen by visual inspection. The beneficial interval is wider than in ImageNet, likely due to the more varied dataset used in the

[3] `https://github.com/Stability-AI/generative-models`

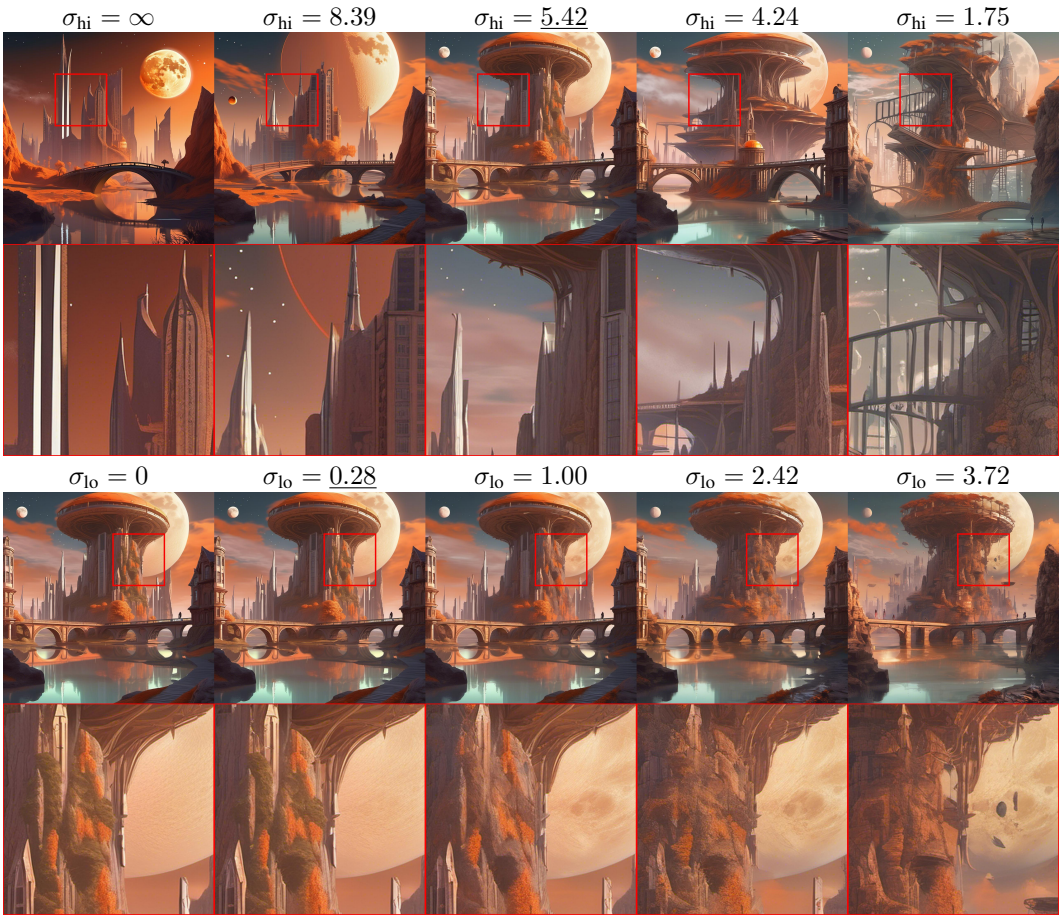


Figure 8: Effect of changing the guidance interval $(\sigma_{\text{lo}}, \sigma_{\text{hi}}]$ with $w = 16$. **Top:** Decreasing σ_{hi}, i.e., disabling guidance at high noise levels, while keeping $\sigma_{\text{lo}} = 0.28$. High values lead to simplified image composition and oversaturated colors (left); low values cause the image to become increasingly convoluted (right). **Bottom:** Increasing σ_{lo}, i.e., disabling guidance at low noise levels, while keeping $\sigma_{\text{hi}} = 5.42$. The value can be made relatively high with no noticeable impact, reducing sampling cost.

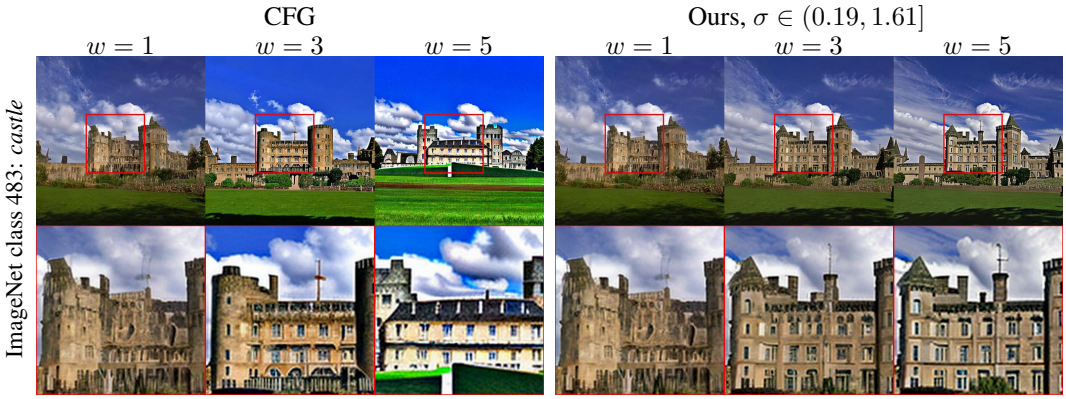


Figure 9: Effect of increasing guidance weight w with CFG vs. our method. **Left:** Increasing the guidance weight with CFG leads to changes in image composition and contrast. **Right:** With our method, increasing w improves image details but retains the overall composition and realistic colors.

training of SD-XL. Consequently, our method leads to over 20% speed-up due to a lower number of unconditional model evaluations [1].

Figure 6 shows a comparison between standard classifier-free guidance with low and high weights (left and middle columns) and our method with high guidance weight (right column). When the guidance weight is increased in standard CFG (middle), the composition of the image tends to change drastically, towards some limited set of per-class "templates". Furthermore, the colors saturate unnaturally as the guidance weight increases. When we limit the guidance interval (right), image diversity is preserved to a significant degree and the color saturation is also reduced, although excessively large guidance weights can still lead to over-saturation.

Figure 7 shows the effect of increasing the guidance weight with our method. With low weight, the images appear blurry, inconsistent, and lacking in detail. Increasing the weight improves the rendition of details while retaining the original image composition.

As the task of selecting the best guidance interval $(\sigma_{\text{lo}}, \sigma_{\text{hi}}]$ with SD-XL is necessarily subjective, we provide a visual ablation of this choice in Figure 8. Modifying the upper limit σ_{hi}, i.e., disabling guidance at high noise levels, has two distinct effects. First, it affects the overall image composition — higher values lead to more simplified image layouts whereas low values lead to unnecessary complexity. Second, high values lead to oversaturated colors whereas lower σ_{hi} leads to a blander color scheme. Similar to EDM2 results, changing the lower limit σ_{lo} has only a modest effect — guidance can be disabled from most of the low noise levels with no noticeable impact while improving the inference speed.

Lastly, Figure 9 compares the effects of increasing the guidance weight in standard CFG vs. our method with EDM2-XXL.

5 Conclusions

Classifier-free guidance is an indispensable tool for improving the results of practically all image-generating diffusion models. As our simple modification improves the results both numerically and visually, and also reduces sampling cost, we recommend exposing the guidance interval as an additional sampler parameter.

Future work could investigate whether the optimal guidance interval can be automatically derived from the ODE, and the role played by the non-idealities in the trained denoiser. A recent work by Biroli et al. [6] predicts from a dataset the interval where the generated images specialize to a certain class. A follow-up study could examine whether their "speciation" interval overlaps with the interval that is beneficial for guidance.

Acknowledgements

This work was partially supported by the European Research Council (ERC Consolidator Grant 866435), and made use of computational resources provided by the Aalto Science-IT project and the Finnish IT Center for Science (CSC).

References

[1] Alex Birch. Turning off classifier-free guidance at low noise levels. Idea mentioned on Twitter. `https://twitter.com/Birchlabs/status/1640033271512702977`, 2023. *Accessed 19 Mar 2024.*

[2] Alex Goodwin. Stable Diffusion dynamic thresholding. GitHub repository. `https://github.com/mcmonkeyprojects/sd-dynamic-thresholding`, 2023. *Accessed 19 Mar 2024.*

[3] AUTOMATIC1111. Stable Diffusion web UI. GitHub repository. `https://github.com/AUTOMATIC1111/stable-diffusion-webui`, 2022. *Accessed 19 Mar 2024.*

[4] Y. Balaji, S. Nah, X. Huang, A. Vahdat, J. Song, K. Kreis, M. Aittala, T. Aila, S. Laine, B. Catanzaro, T. Karras, and M.-Y. Liu. eDiff-I: Text-to-image diffusion models with ensemble of expert denoisers. *CoRR*, abs/2211.01324, 2022.

[5] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo, W. Manassra, P. Dhariwal, C. Chu, Y. Jiao, and A. Ramesh. Improving image generation with better captions. Technical report, OpenAI, 2023.

[6] G. Biroli, T. Bonnaire, V. de Bortoli, and M. Mézard. Dynamical regimes of diffusion models. *CoRR*, abs/2402.18491, 2024.

[7] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proc. CVPR*, 2023.

[8] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, C. Ng, R. Wang, and A. Ramesh. Video generation models as world simulators. Blog post, OpenAI, 2024.

[9] A. Castillo, J. Kohler, J. C. Pérez, J. P. Pérez, A. Pumarola, B. Ghanem, P. Arbeláez, and A. Thabet. Adaptive guidance: Training-free acceleration of conditional diffusion models. *CoRR*, abs/2312.12487, 2023.

[10] H. Chang, H. Zhang, J. Barber, A. Maschinot, J. Lezama, L. Jiang, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, Y. Li, and D. Krishnan. Muse: Text-to-image generation via masked generative transformers. *CoRR*, abs/2301.00704, 2023.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009.

[12] S. Dieleman. Guidance: A cheat code for diffusion models. Blog post. `https://sander.ai/2022/05/26/guidance.html`, 2022.

[13] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *Proc. ICLR*, 2023.

[14] S. Gao, P. Zhou, M.-M. Cheng, and S. Yan. Mdtv2: Masked diffusion transformer is a strong image synthesizer. In *Proc. ICCV*, 2023.

[15] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proc. NIPS*, 2017.

[16] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans. Imagen Video: High definition video generation with diffusion models. *CoRR*, abs/2210.02303, 2022.

[17] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Proc. NeurIPS*, 2020.

[18] J. Ho and T. Salimans. Classifier-free diffusion guidance. In *Proc. NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[19] J. Ho, T. Salimans, A. A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. In *Proc. ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022.

[20] A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.

[21] Jeremy Howard and Rekil Prashanth. Adjusting guidance weight as a function of time. Idea mentioned on Twitter. `https://twitter.com/jeremyphoward/status/1584771100378288129`, 2022. *Accessed 19 Mar 2024.*

[22] T. Karras, M. Aittala, T. Aila, and S. Laine. Elucidating the design space of diffusion-based generative models. In *Proc. NeurIPS*, 2022.

[23] T. Karras, M. Aittala, J. Lehtinen, J. Hellsten, T. Aila, and S. Laine. Analyzing and improving the training dynamics of diffusion models. In *Proc. CVPR*, 2024.

[24] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro. DiffWave: A versatile diffusion model for audio synthesis. In *Proc. ICLR*, 2021.

[25] T. Kynkäänniemi, T. Karras, S. Laine, J. Lehtinen, and T. Aila. Improved precision and recall metric for assessing generative models. In *Proc. NeurIPS*, 2019.

[26] C.-H. Lin, J. Gao, L. Tang, T. Takikawa, X. Zeng, X. Huang, K. Kreis, S. Fidler, M.-Y. Liu, and T.-Y. Lin. Magic3D: High-resolution text-to-3D content creation. In *Proc. CVPR*, 2023.

[27] S. Mahajan, T. Rahman, K. M. Yi, and L. Sigal. Prompting hard or hardly prompting: Prompt inversion for text-to-image diffusion models. In *Proc. CVPR*, 2024.

[28] A. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *Proc. ICML*, 2021.

[29] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proc. ICCV*, 2023.

[30] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *Proc. ICLR*, 2024.

[31] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. In *Proc. ICLR*, 2023.

[32] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov. Grad-TTS: A diffusion probabilistic model for text-to-speech. In *Proc. ICML*, 2021.

[33] A. Raj, S. Kaza, B. Poole, M. Niemeyer, B. Mildenhall, N. Ruiz, S. Zada, K. Aberman, M. Rubenstein, J. Barron, Y. Li, and V. Jampani. DreamBooth3D: Subject-driven text-to-3D generation. In *Proc. ICCV*, 2023.

[34] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proc. CVPR*, 2022.

[35] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proc. CVPR*, 2023.

[36] S. Sadat, J. Buhmann, D. Bradley, O. Hilliges, and R. M. Weber. Cads: Unleashing the diversity of diffusion models through condition-annealed sampling. In *Proc. ICLR*, 2024.

[37] J. R. Shue, E. R. Chan, R. Po, Z. Ankner, J. Wu, and G. Wetzstein. 3D neural field generation using triplane diffusion. In *Proc. CVPR*, 2023.

[38] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proc. ICML*, 2015.

[39] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. In *Proc. ICLR*, 2021.

[40] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. In *Proc. NeurIPS*, 2019.

[41] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *Proc. ICLR*, 2021.

[42] G. Stein, J. C. Cresswell, R. Hosseinzadeh, Y. Sui, B. L. Ross, V. Villecroze, Z. Liu, A. L. Caterini, J. E. T. Taylor, and G. Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Proc. NeurIPS*, 2023.

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

[44] L. Zhang, A. Rao, and M. Agrawala. Adding conditional control to text-to-image diffusion models. In *Proc. ICCV*, 2023.

A Characterizing sampling steps in noise levels

In the main paper, we reported the guidance interval measured in noise levels σ. Here, we show for each model how the indices of sampling steps are mapped to noise levels. For EDM2 models and SD-XL, we use the discretization from [22]. The ith sampling step corresponds to noise level that is given by:

$$\sigma_i = \left(\sigma_{\text{max}}^{\frac{1}{\rho}} + \frac{i}{N-1}\left(\sigma_{\text{min}}^{\frac{1}{\rho}} - \sigma_{\text{max}}^{\frac{1}{\rho}}\right)\right)^{\rho}, \tag{7}$$

where N is the total number of sampling steps, $\sigma_{\text{min}} = 0.002$, $\sigma_{\text{max}} = 80$. With SD-XL, we use $\rho = 3$, which is the default value in the official code, with EDM2 models we use $\rho = 7$. With DiT, we use the iDDPM discretization from [22] which maps the ith sampling step to the corresponding noise level in the following way:

$$\sigma_i = u_{\lfloor j_0 + \frac{M-1-j_0}{N-1} i + \frac{1}{2} \rfloor}, \tag{8}$$

where $u_M = 0$, $u_{j-1} = \sqrt{\frac{u_j^2+1}{\max(\bar{\alpha}_{j-1}/\bar{\alpha}_j, C_1)} - 1}$ and $\bar{\alpha}_j = \sin^2\left(\frac{\pi}{2}\frac{j}{M(C_2+1)}\right)$. We use the default parameters $C_1 = 0.001$, $C_2 = 0.008$, $M = 1000$ and $j_0 = 0$ from [22].

B Additional qualitative results

Figures 10 and 11 show further comparisons between classifier-free guidance and our method. Figures 12 and 13 show additional examples from our method where we increase the guidance weight. Figures 14, 15 and 16 compare classifier-free guidance to our method when the guidance weight is increased.

C Broader impacts

Large-scale diffusion models, such as Stable Diffusion XL, might have various negative societal effects related to the spread of disinformation or amplifying harmful biases and stereotypes. Our method improves the result quality of these models which can potentially further magnify these issues. In the large-scale setting, our method decreases the cost of sampling, but diffusion models continue to require a lot of computing power, which may contribute to wider issues such as climate change.

D Licenses

The pre-trained EDM2 [23] models are licensed under the CC BY-NC-SA 4.0 International License by NVIDIA corporation. The pre-trained SD-XL [30] model is available under the CreativeML Open RAIL++-M License by Stability AI. ImageNet [11] dataset uses a custom non-commercial license.

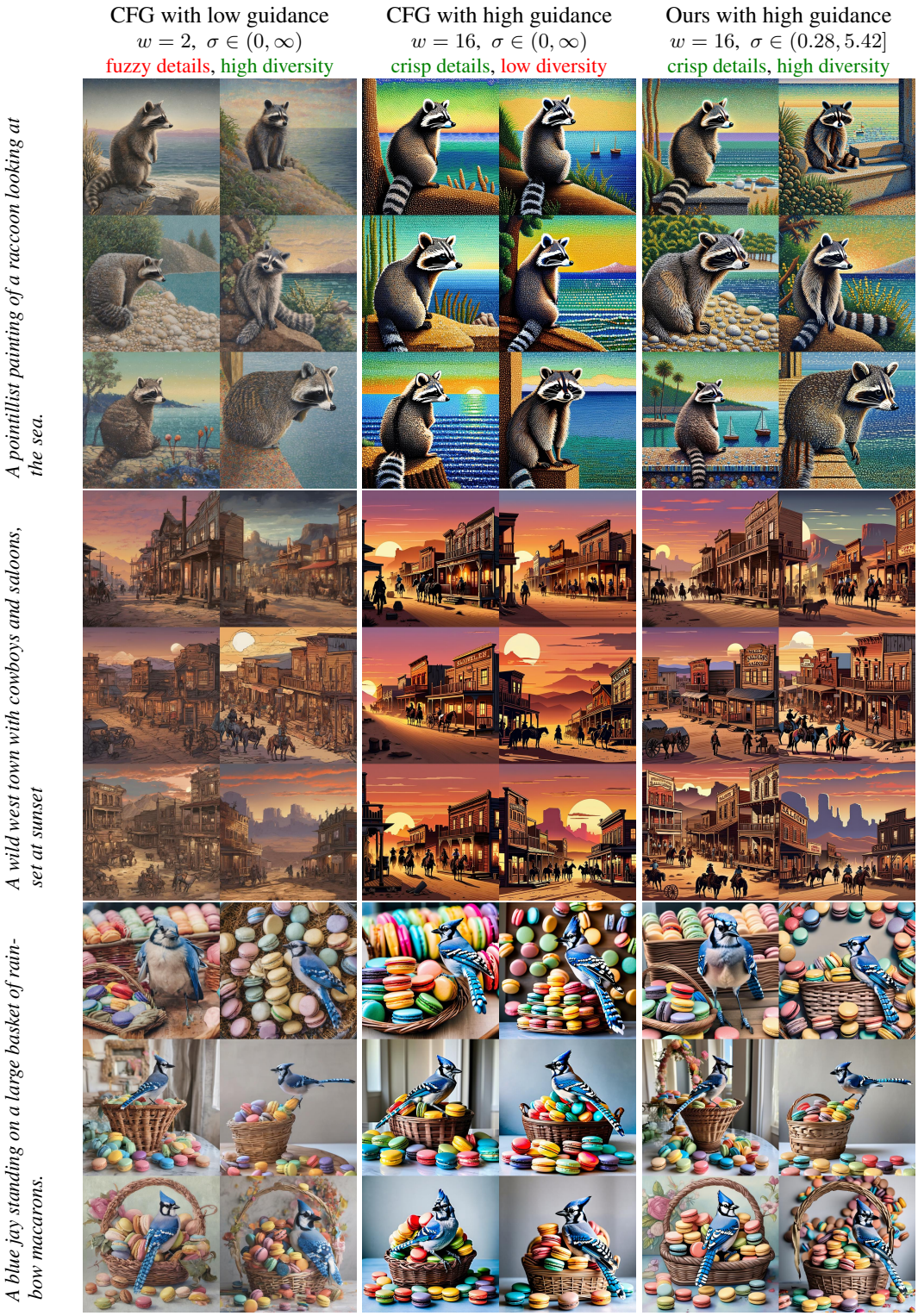
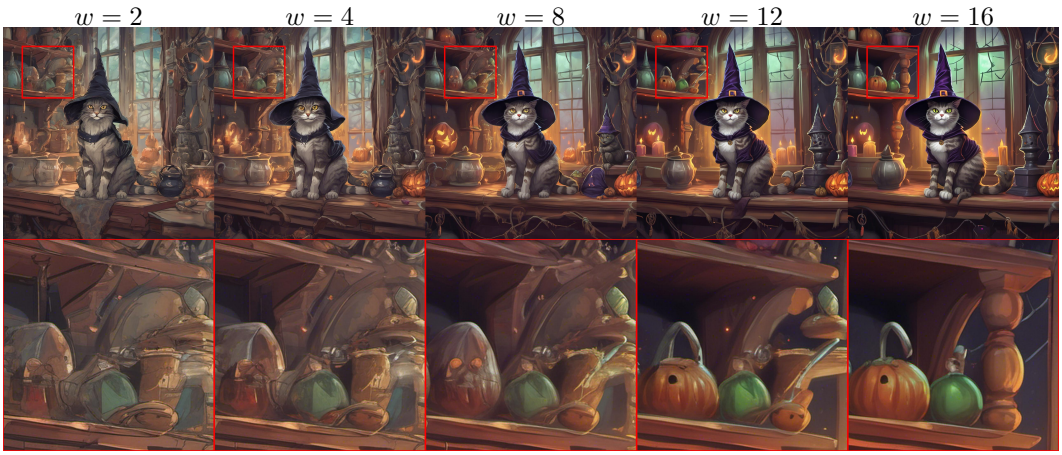


Figure 10: More SD-XL results that demonstrate how CFG with low w yields fuzzy images that lack detail (left) and CFG with high w leads to reduced diversity and oversaturated colors. Our method (right) produces images with crisp details while maintaining natural colors. The degree of the negative effects with CFG varies between prompts.

Figure 11: Additional EDM2-XXL results that demonstrate how CFG with low w yields fuzzy images that lack detail (left) and CFG with high w leads to reduced diversity and oversaturated colors. Our method (right) produces images with crisp details while maintaining natural colors.

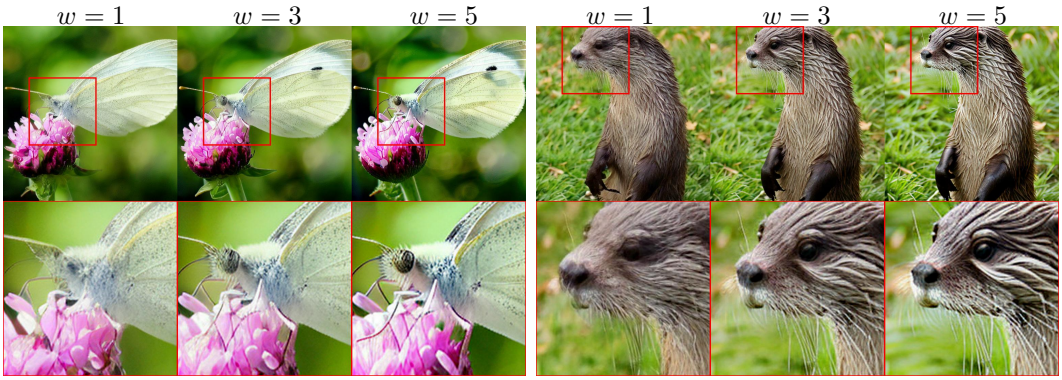


A highly detailed zoomed-in digital painting of a cat dressed as a witch wearing a wizard hat in a haunted house, artstation.

A fantasy landscape of the Shire during sunrise. The Sun is near the horizon and there is fog over farm fields. Highly detailed fantasy art, artstation.

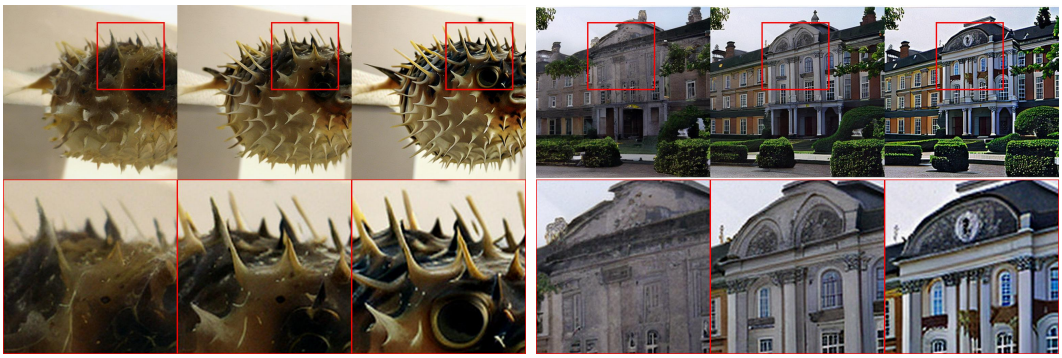
A 4K dslr photo of a hedgehog sitting in a small boat in the middle of a pond. It is wearing a Hawaiian shirt and a straw hat. It is reading a book. There are a few leaves in the background.

Figure 12: More SD-XL results showing the effect of changing w with our method. We limit the guidance to $\sigma \in (0.28, 5.42]$. Increasing w produces images with more well-defined details while maintaining the color palette and the original image composition.

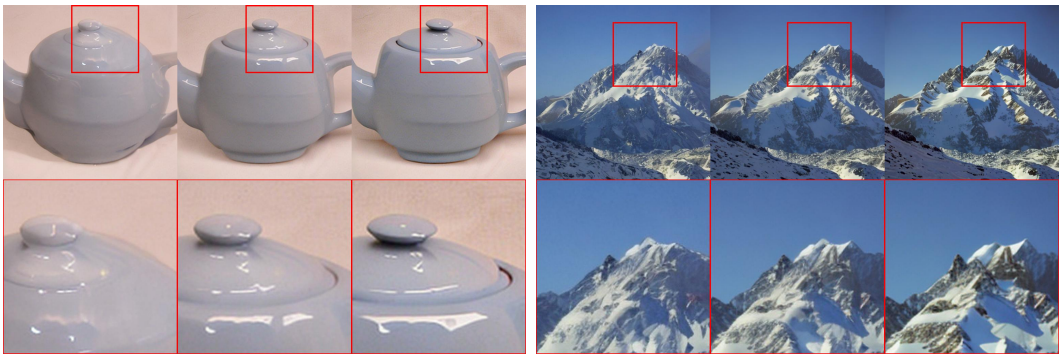


ImageNet class 324: *cabbage butterfly* ImageNet class 360: *otter*

ImageNet class 105: *koala* ImageNet class 555: *fire truck*



ImageNet class 397: *pufferfish* ImageNet class 698: *palace*

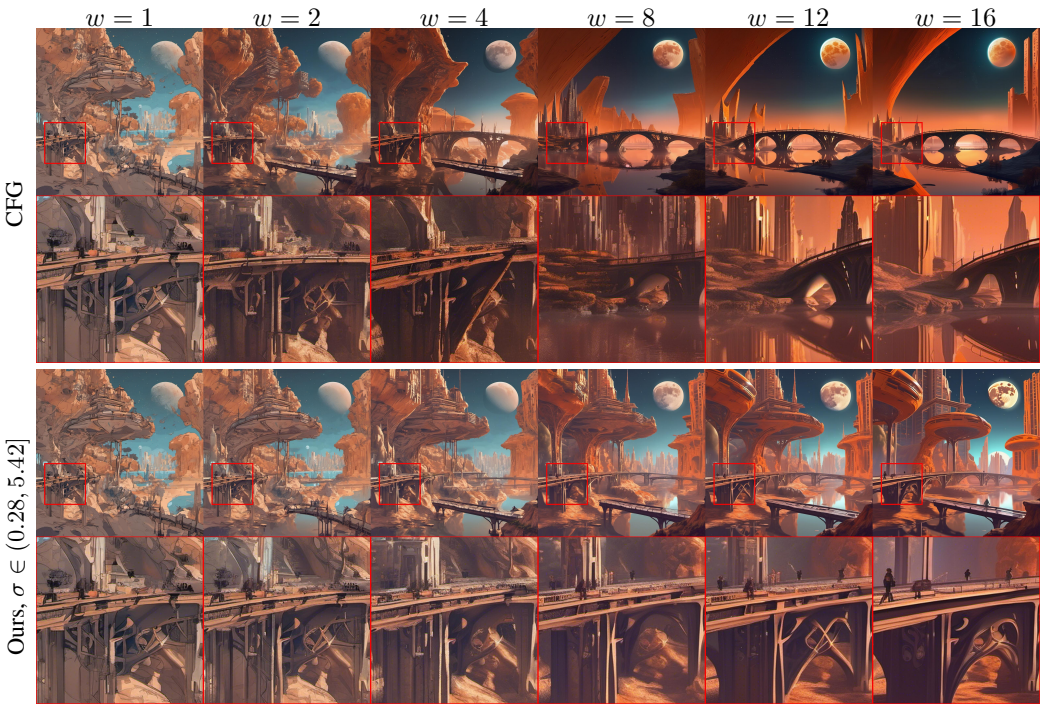


ImageNet class 849: *teapot* ImageNet class 970: *alp*

Figure 13: More EDM2-XXL results showing the effect of changing w with our method. We limit the guidance to $\sigma \in (0.19, 1.61]$. Increasing w produces images with more well-defined details while maintaining the color palette and the original image composition.

A highly detailed paper origami of a Dachshund on a table next to a porcelain teapot, 4k dslr.

Figure 14: Effect of increasing guidance weight w with CFG vs. our method. **Top:** Increasing the guidance weight with CFG leads to large changes in the image composition. Note how the dog's head moves as w changes. **Bottom:** Our method leads to well-defined image details and retains the overall composition to a significant degree.



A fantasy landscape on an alien planet in which there are many buildings. There is a beautiful bridge with a pond in the center. There is one large moon in the sky. The sky is orange. Digital art, artstation

Figure 15: Effect of increasing guidance weight w with CFG vs. our method. **Top:** Increasing the guidance weight with CFG leads to large changes in the image composition. **Bottom:** Our method leads to well-defined image details and retains the overall composition to a significant degree.

Figure 16: Effect of increasing guidance weight w with CFG vs. our method. **Left:** Increasing the guidance weight with CFG leads to large changes in the image composition. **Right:** Our method leads to well-defined image details and retains the overall composition to a significant degree.

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 claims made in the abstract and introduction accurately reflect the experimental results in Section 4.

   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: We discuss the limitations of our method in Section 4.

   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 include 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: Our method is a simple (a few lines of code) modification to the sampling loop and Sections 3 and 4 disclose all the required information to reproduce our results.

   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: The code publicly available.

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, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify all necessary hyperparameters to understand the results in Section 3 and Table 1 reports the specific hyperparameter values to reproduce our results.

   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 show the variation of our method and the baseline in Figure 3 where the shaded regions correspond to the min/max over three evaluations.

   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: The paper does not train any new models. We use pre-trained models that can be run on a desktop computer with a GPU.

   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: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   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 the broader impacts of our work in Appendix C.

    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: [NA]

    Justification: The paper does not release new data or 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: We cite and provide links to the original sources of the pre-trained models used in our work as footnotes in Section 4. The licenses of the models that we used are mentioned in Appendix D.

    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: [NA]

    Justification: The paper does not release new assets.

    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: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    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: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    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.