Supplementary Material:

It's Never Too Late: Noise Optimization for Collapse Recovery in Trained Diffusion Models

1. Implementation Details

1.1. Optimization Objectives and Metrics

Output Diversity. We use multiple diversity objectives that aim at generating a set of diverse images with diffusion models. In the following, we first describe the pairwise diversity metrics that we used.

DINO. This diversity objective and metric uses DINOv2 [18] patch features to measure perceptual diversity as defined in Eq. 3 in the main paper. Specifically, we compute the pairwise cosine distances (i.e. d is the cosine distance) between patch features in different images. Lower values indicate similar images, and values closer to 1 represent higher diversity. We also refer to this metric as "Output variation (DINO)".

DreamSim. We use pairwise DreamSim dissimilarity scores obtained with a DINO ViT-B/16 backbone that was trained to align with human perception [8]. Lower values indicate similar images, whereas values closer to 1 correspond to more diversity in the outputs.

LPIPS. We use LPIPS [29] to quantify the dissimilarity between a pair of images with a VGG [25] backbone. Specifically, LPIPS computes a weighted sum of perceptual similarities across the outputs of all five convolutional blocks of VGG16. Values close to 0 indicate similar images, whereas values closer to 1 indicate higher diversity.

Color Histogram. We consider the pairwise color histogram distance between images. In particular, we calculate color histograms for each channel considering 32 bins. We use soft histograms with Gaussian kernels to ensure that this operation is differentiable. We then measure the pairwise L2 distance between the resulting color histograms of two images, and normalize this such that the final score is in the range $[0, 1]$.

L2. Inspired by the image similarity used in [27], we use a low-resolution L2 distance between pairs of images. In particular, we resize the generated images to 32×32 and compute the L2 distance between the resulting 3072-dimensional vectors representing each image. We normalize this score to be in the range $[0, 1]$. Higher values correspond to higher diversity.

In addition to the above described averaged pairwise diversity objectives, we consider two set-based metrics.

DPP. We normalize the DINOv2 [CLS] token embeddings $\bar{f}i$ for each image $x^{(i)}$. The normalized embeddings are used to construct a similarity kernel matrix $K_s = \bar{F}\bar{F}^T$ where $\bar{F} = [\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_N]^T$, and N the number of images. The kernel is symmetrized as $K_{sym} = (K_s + K_s^T)/2$ and augmented with $K \leftarrow K_{sym} + \epsilon I$ where $\epsilon = 10^{-6}$. The Determinantal Point Process (DPP) score [14] is then computed as the log-determinant:

$$\mathcal{D}_{\text{DPP}} = \log\det(I + K). \tag{1}$$

This score ranges between $[0, \log(16)]$ for a set of four images, with 0 indicating that all images are identical, and 2.77 stating that all images in the set are maximally diverse.

Vendi. Starting with the same similarity kernel K as in DPP, we compute its eigenvalue decomposition to obtain $\lambda_1, \lambda_2, \ldots, \lambda_N$. These eigenvalues are normalized to form a probability distribution $p_i = \lambda_i / \sum_{j=1}^{N} \lambda_j$. The Vendi score [7] is defined as the exponential of the Shannon entropy of this distribution:

$$\mathcal{D}_{\text{Vendi}} = \exp\left(-\sum_{i=1}^{N} p_i \log(p_i + \delta)\right), \tag{2}$$

where $\delta = 10^{-12}$ to prevent numerical issues. This score is between $[1, 4]$ for a set of four images, measuring the effective number of diverse images in the set. A score of 1 signifies that all images are effectively similar, and 4 shows that each image in the set is unique.

Image Quality. We optimize image quality using CLIPScore and a human preference score.

CLIPScore. Similar to [6], we use a reward model that pushes the optimization process to preserve image quality and prompt relevance. Specifically, we use a pretrained

CLIP [22] ViT-B/32 model. [21] also used this model to ensure image quality and prompt following.

HPSv2. We use the HPSv2 [28] metric as another image reward to maintain quality during optimization. It is based on a CLIP [22] ViT-H/14 backbone.

For evaluation, in addition to measuring CLIPScore and HPSv2 we report PickScore [13], and for completeness we also report FID [10] on a subset of results.

PickScore. We reserve PickScore [13] as an independent quality metric since we do not use it during optimizations. The metric uses a CLIP [22] ViT-H/14 backbone fine-tuned on Pick-a-Pic user preference dataset. Higher PickScore values indicate better quality.

FID. For additional comparisons, we also evaluate quality with Fréchet Inception Distance (FID) [10]. To obtain scores, we compare generated images to the COCO [17] validation set which consists of 5000 images. Specifically, we compute FID using the clean-fid library [20] with CLIP ViT-B/32 [22] features and its recommended "clean" preprocessing pipeline. For each prompt in an evaluation set, we generate four samples. In our setting, images are generated from GenEval and DPG prompts that bear little resemblance to COCO images, making the reference distribution a suboptimal match. Furthermore, with only small image sets (e.g. 2212 generated images for GenEval), FID estimates are unreliable [1]. We therefore mainly rely on per-image quality metrics (HPSv2, PickScore) that do not assume a matching reference distribution.

1.2. Hyperparameter Choices

We use the SDXL-Turbo [24], SANA-Sprint [3], PixArt-α-DMD [2], and Flux.1 [schnell] [16] models in our experiments. For the majority of our experiments we show results for batched optimization. In this case for **i.i.d. samples**, we randomly sample input noise and generate a set of four images in a model's default configuration without altering the four initial noises.

For sequential optimization (see main paper Fig. 3), we use 25 iterations, a learning rate of 3.0, $\lambda_{div} = 15$ for the DPP diversity objective, $\lambda_q = 1$ for a HPSv2 quality reward, and gradient clipping of 0.15.

All batched experiments use $\lambda_{\text{reg}} = 0.01$ (Eq. 2 main paper) and pink noise exponent $\alpha = 0.2$ unless otherwise noted. Table 1 summarizes per-experiment settings; Table 2 summarizes settings used in the diversity objective comparison (Sec. 4.1 main paper). For SDXL-Turbo, PixArt, and Flux.1 [schnell], we use image resolutions of 512×512, and 768×768 for SANA-Sprint.

Parmar et al. [21]. We apply [21] to the SDXL-Turbo, SANA-Sprint, PixArt-α, and Flux.1 [schnell] models. We use the default parameters that were used for Flux.1 [schnell] [15, 16] in [21], since this setting is closest to our setup with one-step / few-step models. However, for SDXL-Turbo and PixArt, we use image resolutions of 512×512, 768×768 for SANA-Sprint, and 512×512 for Flux.1 [schnell].

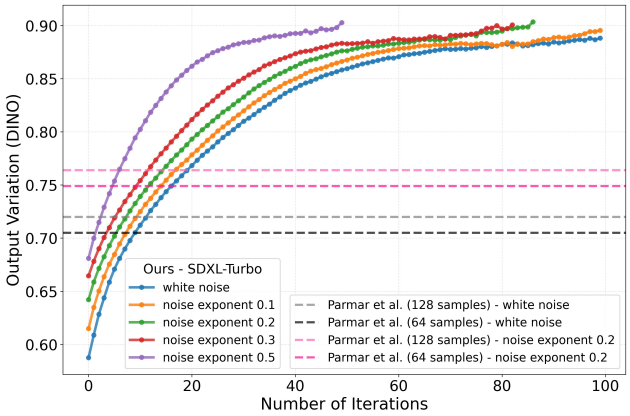


Figure 1. Output variation across optimization iterations for SDXL-Turbo with different noise initializations on GenEval. Higher noise exponents produce greater diversity. Dashed lines are baseline scores from [21] for white noise (gray/black) and pink noise with exponent 0.2 (pink tones) using 64 and 128 samples. Our approach reaches higher diversity (output variation) than [21], requiring only relatively few iterations to outperform [21].

1.3. Datasets

GenEval [9] is a text-to-image generation benchmark that evaluates models across 553 diverse prompts requiring understanding of complex compositional relationships. Unless mentioned otherwise, we report results across all prompts in the dataset.

T2I-CompBench [19] tests compositional understanding in text-to-image models across eight distinct categories: color, shape, texture, spatial relationships, non-spatial attributes, complex compositions, 3D spatial reasoning, and numeracy. We select 50 random prompts per category, resulting in a set of 400 prompts.

DPG-Bench [11] evaluates image generation on 1,065 long, detailed prompts with an average length of 67 words. We use this to assess our diversity optimization on highly specified text prompts.

2. Computational Cost

We measure the time per iteration on a single A100 80GB GPU in Tab. 3. Numbers reported are an average over 100 iterations with the error reported over three different seeds. It takes less than 15 iterations to reach similar levels of diversity as Parmar et al. [21] on GenEval [9] (see Fig. 1).

Table 1. Hyperparameters per experiment. λ_{div} and λ_q weight the diversity and quality terms in Eq. 2 (main paper). "Revert" indicates whether optimization reverts to the previous latent when HPSv2 drops below a threshold. For Flux.1 [schnell], white noise used no HPSv2 weighting, and pink noise used no HPSv2 weighting until iteration 20.

Table / Fig.	Model	Noise	Objective	λ_{div}	λ_q	LR	Grad Clip	Iter.	Revert
Tab. 1, 3, 4	SDXL-Turbo	white / pink	DINO + CLIP	80	50	10.0	0.1	100	–
Tab. 3, 4	PixArt-α	white / pink	DINO + CLIP	80	50	10.0	0.1	100	–
Tab. 3, 4	SANA-Sprint-1.6B	white / pink	DINO + CLIP	25	10	10.0	0.1	100	–
Tab. 2	SDXL-Turbo	white	DPP + HPSv2	150	3	6.0	0.1	150	–
Tab. 2	SDXL-Turbo	pink	DPP + HPSv2	150	3	6.0	0.1	150	–
Tab. 2	Flux.1 [schnell]	white	DPP + HPSv2	1.5	–	6.0	0.1	80	hard < 0.31
Tab. 2	Flux.1 [schnell]	pink	DPP + HPSv2	5	5	2	0.15	80	relative < 0.05
Sequential generation									
Fig. 3	Flux.1 [schnell]	white	DPP + HPSv2	15	1	3.0	0.15	25	–

Table 2. (SDXL-Turbo, white noise, GenEval). $\lambda_q = 10$, LR 10.0, and grad. clip 0.1 for all objectives.

Objective	λ_{div}	Max Iter.	Stop Threshold	Threshold Type
DINO	50	100	0.9	absolute
DreamSim	70	50	0.9	absolute
LPIPS	60	60	0.9	absolute
Color Hist.	60	60	$4\times$	relative to i.i.d.
L2	60	60	$2\times$	relative to i.i.d.
DPP	50	100	$4\times$	relative to i.i.d.
Vendi	50	100	$4\times$	relative to i.i.d.

Table 3. Time per iteration of our proposed optimization approach. We report time on a single A100 80GB in seconds using DPP and HPSv2 objectives.

Model	Time per Iteration
SDXL-Turbo	$0.345_{\pm 0.004}$
Flux.1 [schnell]	$1.092_{\pm 0.008}$

3. Additional Experimental Results

More Model Comparisons. We provide additional results for PixArt-α and SANA-Sprint-1.6B. We evaluate models on GenEval [9] and a subset of 50 prompts per category of T2I-CompBench [19]. Across models, we observe substantial diversity gains with minimal loss in image quality, as reported in Tab. 4 and Tab. 5. To obtain our results on PixArt-α, SANA-Sprint-1.6B, we use the hyperparameters specified in Sec. 1.2. For the SDXL-Turbo results, we use the same settings as Tab. 1 in the main paper. For comparison to [21], we optimize using DINOv2 [18] and CLIP [22] across a batch of 4 output images. All pink noise results are obtained with $\alpha = 0.2$.

Results for Different Diversity Objectives. For the SDXL-Turbo model, we additionally evaluate the effect of different diversity objectives during optimization (Tab. 7). Using the same hyperparameters in Tab. 2, we report scores optimizing for diversity with DINOv2 [18], DreamSim [8], LPIPS [29], Color Histogram [27], L2 distance, DPP [5], and Vendi [7]. For quality we use a CLIP objective. We observe that set-level objectives DPP and Vendi produce the best diversity scores, consistent with our user study in Fig. 6 of the main text.

We show generation results that compare different diversity objectives in Fig. 13 and Fig. 14. These visualizations correspond to the quantitative results in Tab. 7. We can observe that our approach yields more diverse image output sets compared to [21] and generations from i.i.d.-sampled noise initializations across different diversity objectives. All generations are obtained from white noise initializations using the SDXL-Turbo model.

Qualitative Examples for SDXL-Turbo. In Fig. 2, we show example generations from SDXL-Turbo from the experiments in Tab. 4 and Tab. 5. We observe that our method produces greater visual diversity in terms of color, lighting, and pose across all prompts. We further observe that using pink noise initialization improves diversity even under i.i.d. sampling and [21]. Example generations for different diver-

Table 4. Output diversity and image-text alignment results on GenEval and T2I-CompBench for our proposed method with the PixArt-α, SANA-Sprint-1.6B, and SDXL-Turbo models using white noise initialization. Output diversity is measured with averaged pairwise DINO, DreamSim, and LPIPS scores.

	GenEval [9]				**T2I-CompBench** [12]			
Method	DINO	DreamSim	LPIPS	CLIPScore	DINO	DreamSim	LPIPS	CLIPScore
PixArt-α [2]								
i.i.d.	$0.431_{\pm 0.094}$	$0.182_{\pm 0.080}$	$0.474_{\pm 0.119}$	$0.326_{\pm 0.030}$	$0.469_{\pm 0.084}$	$0.188_{\pm 0.069}$	$0.512_{\pm 0.099}$	$0.326_{\pm 0.027}$
Parmar et al. [21]	$0.559_{\pm 0.091}$	$0.246_{\pm 0.094}$	$0.569_{\pm 0.107}$	$0.327_{\pm 0.028}$	$0.590_{\pm 0.078}$	$0.256_{\pm 0.088}$	$0.593_{\pm 0.088}$	$0.328_{\pm 0.027}$
Ours (DINO)	$0.695_{\pm 0.063}$	$0.335_{\pm 0.107}$	$0.664_{\pm 0.089}$	$0.337_{\pm 0.026}$	$0.716_{\pm 0.060}$	$0.331_{\pm 0.102}$	$0.674_{\pm 0.072}$	$0.335_{\pm 0.023}$
SANA-Sprint-1.6B [3]								
i.i.d.	$0.526_{\pm 0.088}$	$0.229_{\pm 0.075}$	$0.635_{\pm 0.087}$	$0.336_{\pm 0.032}$	$0.562_{\pm 0.074}$	$0.252_{\pm 0.078}$	$0.656_{\pm 0.066}$	$0.334_{\pm 0.029}$
Parmar et al. [21]	$0.714_{\pm 0.060}$	$0.354_{\pm 0.095}$	$0.741_{\pm 0.055}$	$0.342_{\pm 0.032}$	$0.684_{\pm 0.060}$	$0.331_{\pm 0.089}$	$0.718_{\pm 0.049}$	$0.338_{\pm 0.028}$
Ours (DINO)	$0.744_{\pm 0.061}$	$0.438_{\pm 0.099}$	$0.781_{\pm 0.062}$	$0.335_{\pm 0.030}$	$0.738_{\pm 0.056}$	$0.437_{\pm 0.105}$	$0.767_{\pm 0.053}$	$0.330_{\pm 0.029}$
SDXL-Turbo [24]								
i.i.d.	$0.588_{\pm 0.083}$	$0.249_{\pm 0.089}$	$0.642_{\pm 0.059}$	$0.335_{\pm 0.031}$	$0.586_{\pm 0.079}$	$0.244_{\pm 0.077}$	$0.634_{\pm 0.056}$	$0.332_{\pm 0.029}$
Parmar et al. [21]	$0.705_{\pm 0.065}$	$0.331_{\pm 0.098}$	$0.682_{\pm 0.055}$	$0.333_{\pm 0.028}$	$0.701_{\pm 0.063}$	$0.329_{\pm 0.087}$	$0.680_{\pm 0.048}$	$0.334_{\pm 0.029}$
Ours (DINO)	$0.784_{\pm 0.026}$	$0.411_{\pm 0.102}$	$0.767_{\pm 0.052}$	$0.349_{\pm 0.029}$	$0.799_{\pm 0.021}$	$0.424_{\pm 0.085}$	$0.764_{\pm 0.056}$	$0.351_{\pm 0.027}$

Table 5. Output diversity and image-text alignment results on GenEval and T2I-CompBench for our proposed method and pink noise initialization with the PixArt-α, SANA-Sprint-1.6B, and SDXL-Turbo models. Output diversity is measured with averaged pairwise DINO, DreamSim, and LPIPS scores.

		GenEval [9]				**T2I-CompBench** [12]			
Method	**Noise**	DINO	DreamSim	LPIPS	CLIPScore	DINO	DreamSim	LPIPS	CLIPScore
PixArt-α [2]									
i.i.d.	🅟	$0.533_{\pm 0.088}$	$0.244_{\pm 0.091}$	$0.604_{\pm 0.116}$	$0.326_{\pm 0.030}$	$0.558_{\pm 0.077}$	$0.247_{\pm 0.083}$	$0.626_{\pm 0.095}$	$0.325_{\pm 0.027}$
Parmar et al. [21]	🅟	$0.664_{\pm 0.074}$	$0.319_{\pm 0.104}$	$0.684_{\pm 0.094}$	$0.323_{\pm 0.029}$	$0.675_{\pm 0.066}$	$0.326_{\pm 0.095}$	$0.692_{\pm 0.077}$	$0.324_{\pm 0.026}$
Ours (DINO)	🅟	$0.764_{\pm 0.039}$	$0.388_{\pm 0.102}$	$0.750_{\pm 0.067}$	$0.335_{\pm 0.029}$	$0.770_{\pm 0.046}$	$0.377_{\pm 0.097}$	$0.748_{\pm 0.063}$	$0.333_{\pm 0.024}$
SANA-Sprint-1.6B [3]									
i.i.d.	🅟	$0.551_{\pm 0.083}$	$0.235_{\pm 0.075}$	$0.649_{\pm 0.083}$	$0.335_{\pm 0.033}$	$0.584_{\pm 0.069}$	$0.259_{\pm 0.079}$	$0.670_{\pm 0.065}$	$0.334_{\pm 0.029}$
Parmar et al. [21]	🅟	$0.737_{\pm 0.053}$	$0.369_{\pm 0.093}$	$0.767_{\pm 0.050}$	$0.341_{\pm 0.032}$	$0.705_{\pm 0.056}$	$0.346_{\pm 0.090}$	$0.736_{\pm 0.048}$	$0.335_{\pm 0.028}$
Ours (DINO)	🅟	$0.753_{\pm 0.049}$	$0.440_{\pm 0.093}$	$0.784_{\pm 0.056}$	$0.334_{\pm 0.031}$	$0.750_{\pm 0.046}$	$0.443_{\pm 0.096}$	$0.773_{\pm 0.050}$	$0.330_{\pm 0.030}$
SDXL-Turbo [24]									
i.i.d.	🅟	$0.642_{\pm 0.068}$	$0.305_{\pm 0.090}$	$0.729_{\pm 0.052}$	$0.328_{\pm 0.031}$	$0.643_{\pm 0.071}$	$0.303_{\pm 0.080}$	$0.719_{\pm 0.055}$	$0.326_{\pm 0.028}$
Parmar et al. [21]	🅟	$0.749_{\pm 0.054}$	$0.392_{\pm 0.100}$	$0.757_{\pm 0.048}$	$0.323_{\pm 0.028}$	$0.742_{\pm 0.055}$	$0.391_{\pm 0.088}$	$0.751_{\pm 0.049}$	$0.328_{\pm 0.027}$
Ours (DINO)	🅟	$0.786_{\pm 0.028}$	$0.427_{\pm 0.095}$	$0.811_{\pm 0.044}$	$0.341_{\pm 0.029}$	$0.804_{\pm 0.026}$	$0.440_{\pm 0.084}$	$0.808_{\pm 0.049}$	$0.344_{\pm 0.026}$

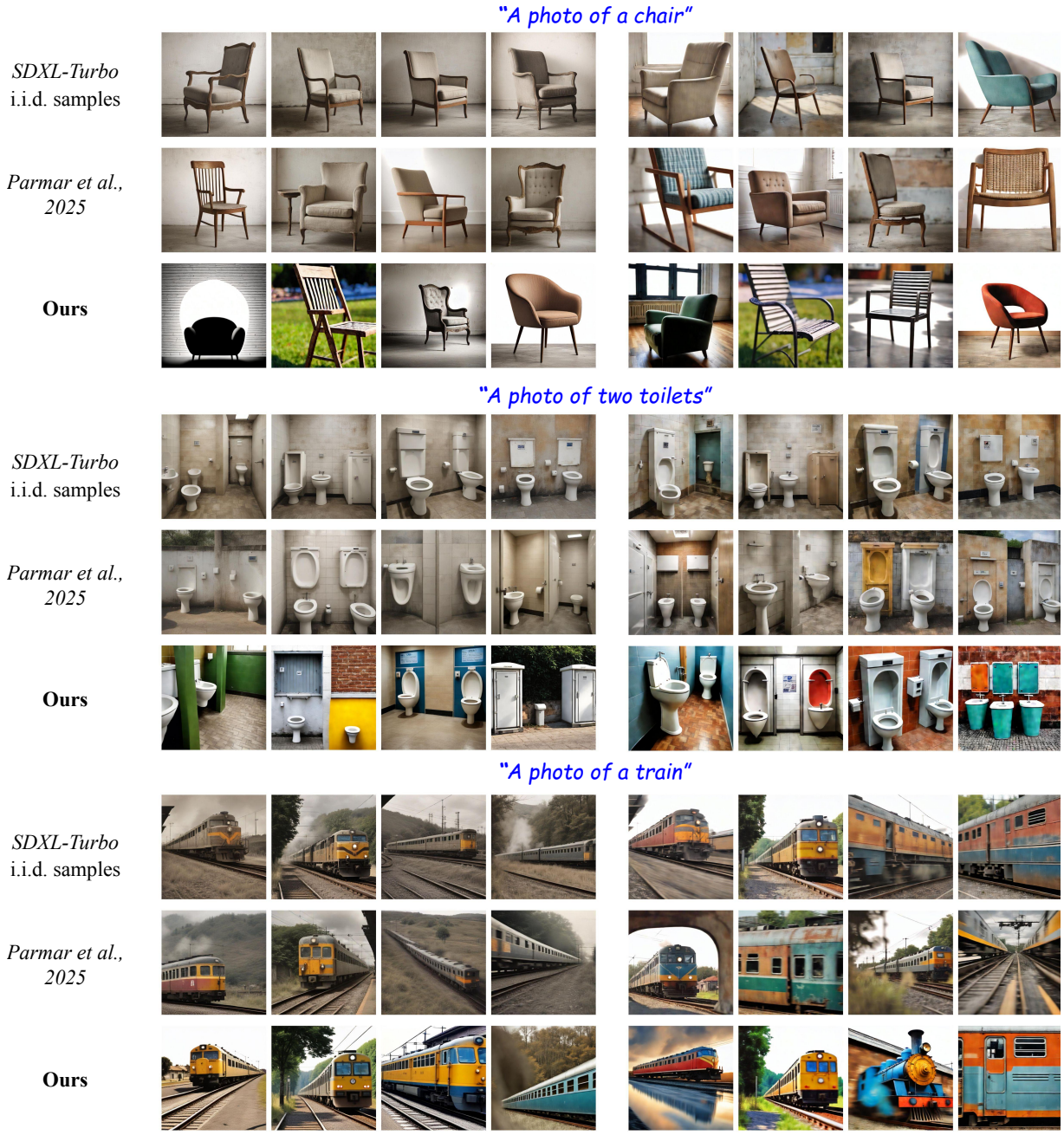
sity objectives can be found in Fig. 5 of the main text. We also provide additional example generations using DPP and HPSv2 objectives for white and pink noise initializations in Fig. 3.

Qualitative Results for Flux.1 [schnell]. We additionally test our optimization on a larger model, Flux.1 [schnell]. Using the best diversity objective from our ablations DPP, we generate results in Fig. 5. Compared to i.i.d. sampling and the default settings from [21], we observe greater output diversity across multiple prompts, particularly in terms of object color, orientation, lighting, and also different backgrounds and positioning. We also provide additional examples optimizing with pink noise initialization in Fig. 6.

In Fig. 4, we demonstrate that our method can be scaled to larger image sets such as 16 generations even on larger models like Flux.1 [schnell]. Compared to i.i.d. sampling, we again see greater diversity across different text prompts. Here, we use 25 iterations, a learning rate of 3.0, $\lambda_{div} = 15$ for the DPP diversity objective, $\lambda_q = 1$ for a HPSv2 quality reward, and gradient clipping of 0.15.

Quantitative Comparisons for Flux.1 [schnell]. For longer complex prompts, we provide quantitative results on DPG-Bench [11]. We evaluate Flux.1 [schnell] in Tab. 6 and again demonstrate that our method improves diversity scores across multiple metrics. We use the same hyperparameters as Tab. 2 in the main paper.

We also provide additional baseline comparisons to guidance-based methods such as Particle Guidance [4], CAD [23], and NegToMe [26] (Tab. 8). In line with [21], we observe that [4] does not significantly improve diver-



a) White noise initializations b) Pink noise initializations

Figure 2. Image generations using our noise optimization approach for SDXL-Turbo yields improved diversity within generated image sets compared to i.i.d sampling and [21]. Pink noise initializations (b) give more diverse generations than standard white noise (a). Ours uses the DINO diversity objective (similar to Tab. 4 and Tab. 5).

sity. These methods all require multi-step models, so we use Flux.1 [schnell]. [23] and [26] are more effective, but their quality-diversity trade-off results in worse image quality for higher diversity.

4. Quality-Diversity Relationship

The scatter plot in Fig. 7 illustrates the relationship between image quality (measured by CLIPScore) and output diversity (DINO) throughout the optimization process for the white noise configuration on the GenEval dataset. The plot corresponds to the setup used for Fig. 1. Note that

Figure 3. Diverse image generation with SDXL-Turbo using white and pink noise initialization with DPP [5] and HPSv2 [28] objectives. We observe that our method improves diversity compared to i.i.d. sampling with white noise. In addition, using pink noise simply at inference time without any optimization increases diversity for both i.i.d. sampling and [21].

Table 6. Output diversity on DPG-Bench [11] with Flux.1 [schnell] [16] using white initial noise optimized with DPP [5] diversity objective.

Method	DreamSim	Vendi (DINO)	HPSv2	PickScore	FID
i.i.d.	$0.197_{\pm 0.062}$	$1.787_{\pm 0.358}$	$0.278_{\pm 0.036}$	$0.217_{\pm 0.012}$	22.458
Ours	$0.285_{\pm 0.077}$	$2.319_{\pm 0.474}$	$0.270_{\pm 0.033}$	$0.215_{\pm 0.011}$	21.468

early stopping terminated optimization after 100 iterations or when the DINO diversity objective reached a threshold of 0.9.

Each point in the plot represents a single iteration across all prompts, colored by the percentage of total iterations completed (darker points indicate early iterations, lighter points indicate later stages). The black line shows the averaged trajectory across all prompts, revealing that both CLIPScore and DINO diversity increase jointly during optimization. This demonstrates that our approach overcomes the quality-diversity tradeoff described in [21]. Our improved output variation does not come at the expense of prompt alignment.

5. Noise Evolution Analysis

Here, we provide further analysis of the change in noise latents across iterations. In Fig. 10, we show the average noise change on the raw noise signal, measured by the L2 norm. The shaded regions around the lines indicate the standard deviation, showing the variability in noise change across different samples. We observe that the L2 norm increases steadily over iterations for white noise initializations.

The average norm change for white noise initializations is slightly lower for pink noise compared to white noise (Fig. 10). This confirms that using pink noise as initialization is favorable for our optimization.

We also analyze the spatial change in noise, both in general and decomposed into frequency bands (Figs. 8 and 9) for SDXL-Turbo. The first column in Fig. 8 shows the images produced from randomly sampled white noise initializations. Subsequent columns show the intermediate outputs, with the final column displaying the images after optimization. For each iteration, we also visualize a heatmap of the noise change, computed as the averaged L2 difference between the current latent and its initial value. Early in the process the heatmaps remain dark, indicating minimal deviation from the original noise. As optimization proceeds, brighter regions emerge in areas where the noise undergoes substantial modification. These regions align with the parts of the image that change the most (e.g. altered bird species or rearranged branches).

Furthermore, we visualize the noise evolution decomposed into frequency bands in Fig. 9. This visualization demonstrates that the low frequency components of the noise are being modified most significantly during the optimization process.

Noise Delta Computation. For each optimization step t, let $\mathbf{z}_t \in \mathbb{R}^{N\times C\times H\times W}$ be the noise. We define the noise change as $\Delta\mathbf{z}_t = \mathbf{z}_t - \mathbf{z}_{t-1}$, with $\mathbf{z}_0$ the initial noise. To visualize how the noise changes spatially, we compute

$$M_t(h,w) = \sqrt{\sum_{c=1}^{C} (\Delta\mathbf{z}_t)^2_{c,h,w}}. \tag{3}$$

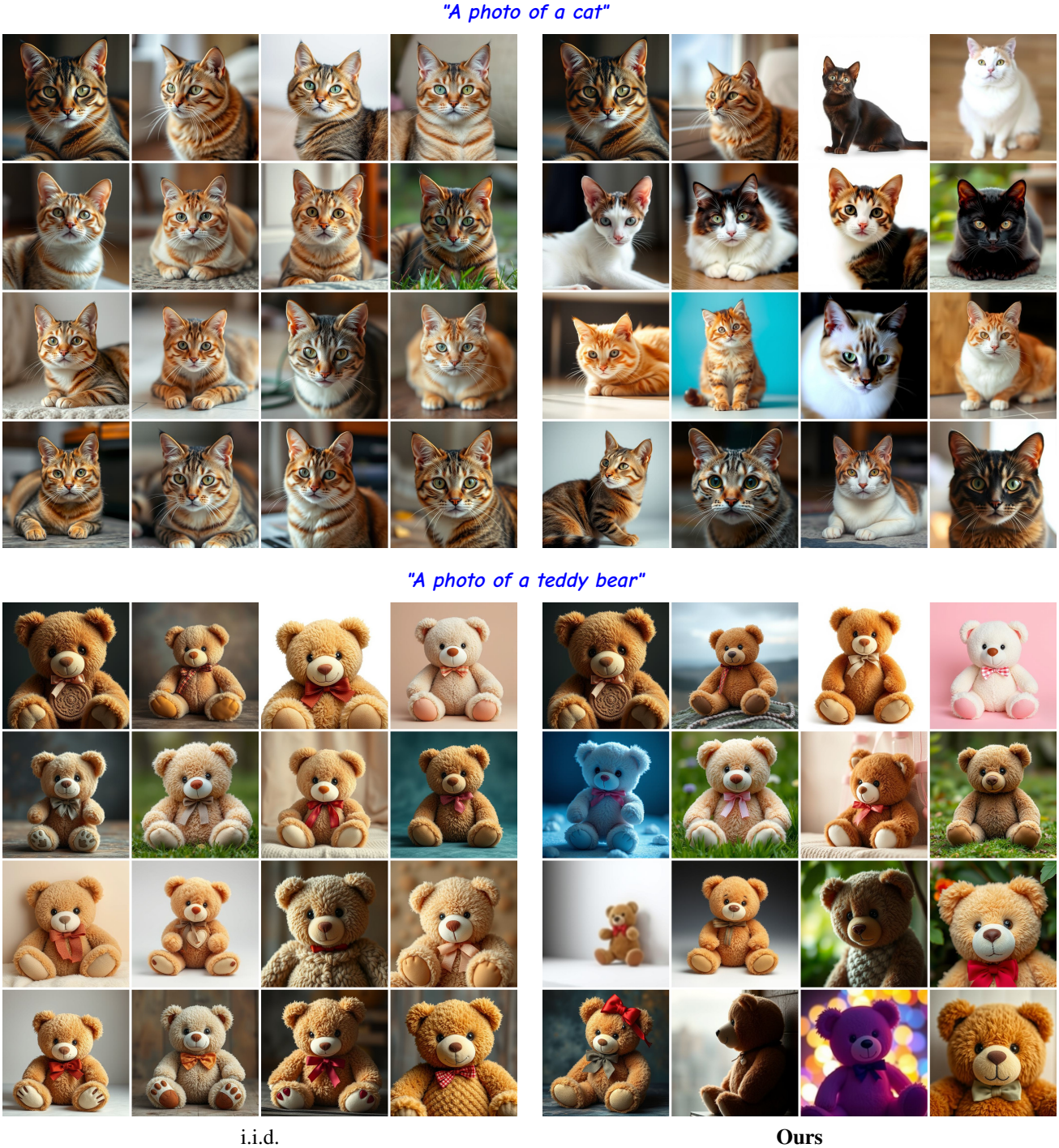


Figure 4. Our method scales to large, diverse image sets via sequential generation. For Flux.1 [schnell], our optimization yields improved diversity of generated image sets compared to i.i.d sampling and scales to larger sets such as the 16 shown here.

This results in a heatmap $M_t \in \mathbb{R}^{H \times W}$ showing the noise change at each location.

Frequency Band Decomposition. We decompose M_t into three frequency bands. For this, we compute the 2D FFT:

$$\mathcal{F}_t(u, v) = \mathcal{F}\{M_t\}, \quad P_t(u, v) = |\mathcal{F}_t(u, v)|^2,$$

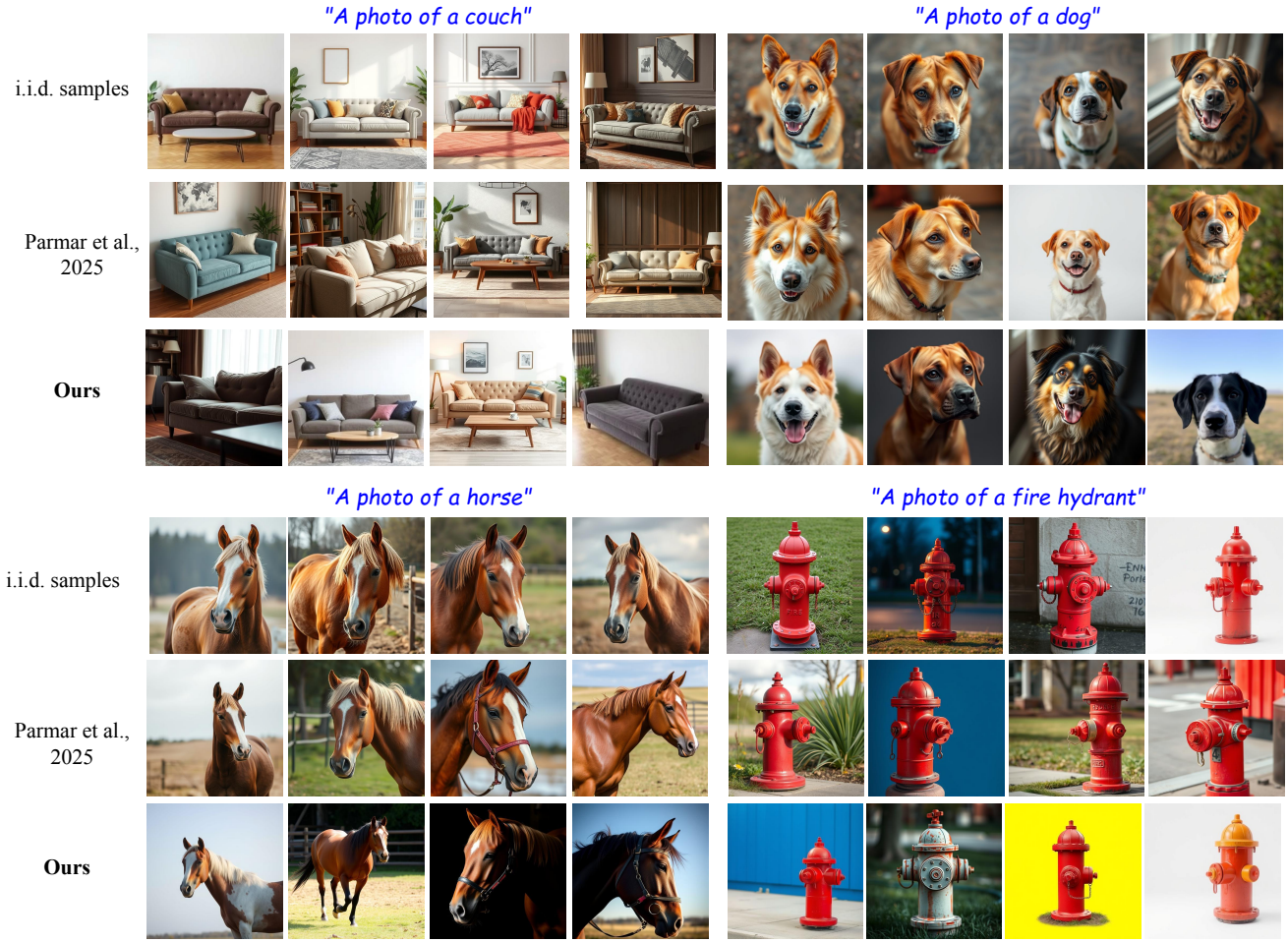


Figure 5. Image generations applying our method to Flux.1 [schnell] [16] with white noise initialization. We achieve greater visual diversity compared to baselines while maintaining image quality.

Figure 6. Image generations applying our method to Flux.1 [schnell] [16] and SDXL-Turbo [24] with pink noise initialization. We achieve greater visual diversity compared to baselines while maintaining image quality.

Table 7. Impact of different optimization objectives for our pipeline with SDXL-Turbo on GenEval using white noise initializations. Our optimization pipeline does not hurt the overall image quality (HPSv2, CLIPScore, PickScore, FID) across different diversity objectives (the result on the metric that we optimized for is shown in brackets), despite only using a weakly weighted CLIP text-image objective as an additional reward to maintain adherence to the input prompt.

Objective	DINO	DreamSim	LPIPS	Color	L2	DPP	Vendi	HPSv2	CLIPScore	PickScore	FID
None (init)	$0.588_{\pm 0.082}$	$0.249_{\pm 0.089}$	$0.643_{\pm 0.059}$	$0.094_{\pm 0.041}$	$0.279_{\pm 0.046}$	$2.104_{\pm 0.216}$	$1.999_{\pm 0.505}$	$0.263_{\pm 0.027}$	$0.335_{\pm 0.031}$	$0.224_{\pm 0.013}$	24.515
DINO	$(0.892_{\pm 0.049})$	$0.476_{\pm 0.105}$	$0.799_{\pm 0.056}$	$0.165_{\pm 0.057}$	$0.436_{\pm 0.061}$	$2.678_{\pm 0.114}$	$3.652_{\pm 0.368}$	$0.260_{\pm 0.024}$	$0.347_{\pm 0.032}$	$0.219_{\pm 0.012}$	21.802
DreamSim	$0.718_{\pm 0.083}$	$(0.763_{\pm 0.245})$	$0.786_{\pm 0.082}$	$0.177_{\pm 0.068}$	$0.407_{\pm 0.079}$	$2.450_{\pm 0.218}$	$2.919_{\pm 0.613}$	$0.243_{\pm 0.027}$	$0.333_{\pm 0.028}$	$0.216_{\pm 0.013}$	22.760
LPIPS	$0.680_{\pm 0.077}$	$0.383_{\pm 0.119}$	$(0.852_{\pm 0.100})$	$0.146_{\pm 0.062}$	$0.370_{\pm 0.065}$	$2.219_{\pm 0.221}$	$2.276_{\pm 0.552}$	$0.269_{\pm 0.025}$	$0.338_{\pm 0.030}$	$0.223_{\pm 0.011}$	24.170
Color	$0.661_{\pm 0.076}$	$0.401_{\pm 0.117}$	$0.726_{\pm 0.069}$	$(0.376_{\pm 0.156})$	$0.408_{\pm 0.080}$	$2.241_{\pm 0.216}$	$2.330_{\pm 0.552}$	$0.259_{\pm 0.027}$	$0.346_{\pm 0.032}$	$0.215_{\pm 0.014}$	23.756
L2	$0.684_{\pm 0.065}$	$0.362_{\pm 0.091}$	$0.768_{\pm 0.056}$	$0.145_{\pm 0.052}$	$(0.492_{\pm 0.081})$	$2.237_{\pm 0.213}$	$2.318_{\pm 0.538}$	$0.268_{\pm 0.024}$	$0.335_{\pm 0.033}$	$0.208_{\pm 0.012}$	25.686
DPP	$0.787_{\pm 0.043}$	$0.477_{\pm 0.098}$	$0.778_{\pm 0.054}$	$0.170_{\pm 0.061}$	$0.444_{\pm 0.058}$	$(2.772_{\pm 0.000})$	$4.000_{\pm 0.001}$	$0.261_{\pm 0.025}$	$0.368_{\pm 0.035}$	$0.219_{\pm 0.012}$	22.062
Vendi	$0.791_{\pm 0.043}$	$0.486_{\pm 0.103}$	$0.782_{\pm 0.052}$	$0.167_{\pm 0.060}$	$0.440_{\pm 0.057}$	$2.773_{\pm 0.000}$	$(4.000_{\pm 0.000})$	$0.259_{\pm 0.024}$	$0.356_{\pm 0.034}$	$0.219_{\pm 0.017}$	21.925

Table 8. Baseline comparisons to guidance-based methods on GenEval with Flux.1 [schnell]. Methods include Particle Guidance [4], CADS [23], and NegToMe [26].

Method	DreamSim	Vendi (DINO)	HPSv2	PickScore	CLIPScore	FID
i.i.d.	$0.307_{\pm 0.100}$	$2.013_{\pm 0.490}$	$0.304_{\pm 0.025}$	$0.232_{\pm 0.010}$	$0.332_{\pm 0.031}$	27.871
PG [9]	$0.296_{\pm 0.095}$	$2.047_{\pm 0.512}$	$0.304_{\pm 0.024}$	$0.231_{\pm 0.001}$	$0.331_{\pm 0.032}$	27.450
Parmar et al. [21]	$0.399_{\pm 0.104}$	$2.460_{\pm 0.573}$	$0.294_{\pm 0.022}$	$0.228_{\pm 0.009}$	$0.324_{\pm 0.027}$	26.570
CADS [23]	$0.363_{\pm 0.117}$	$2.365_{\pm 0.611}$	$0.295_{\pm 0.028}$	$0.228_{\pm 0.001}$	$0.323_{\pm 0.031}$	25.570
NegToMe [26]	$0.385_{\pm 0.092}$	$2.355_{\pm 0.515}$	$0.291_{\pm 0.022}$	$0.227_{\pm 0.009}$	$0.328_{\pm 0.029}$	26.090
Ours	$0.446_{\pm 0.116}$	$2.753_{\pm 0.587}$	$0.293_{\pm 0.025}$	$0.229_{\pm 0.009}$	$0.329_{\pm 0.029}$	26.703

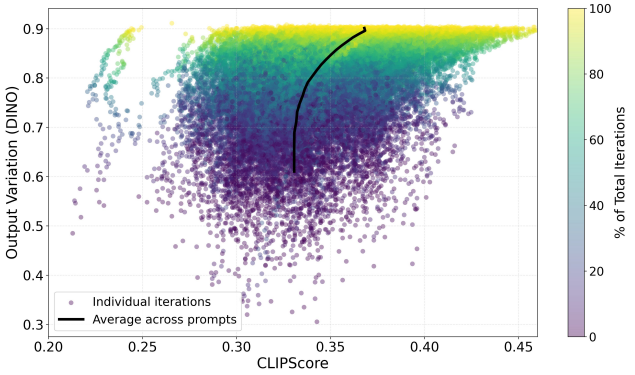


Figure 7. Scatter plot of CLIPScore and DINO diversity during optimization for SDXL-Turbo with white noise initialization on GenEval. Points are colored by iteration progress. The averaged trajectory (black) shows joint improvements in image quality and diversity, demonstrating that our method overcomes the quality–diversity tradeoff.

where (u, v) are frequency coordinates. The radial distance from the zero-frequency center is

$$r(u, v) = \sqrt{(u - u_c)^2 + (v - v_c)^2}, \tag{4}$$

and we define three frequency bins:

$$\begin{aligned} \text{Low: } & [0, r_{\max}/3), \\ \text{Mid: } & [r_{\max}/3, 2r_{\max}/3), \\ \text{High: } & [2r_{\max}/3, r_{\max}], \end{aligned}$$

for $r_{\max} = \sqrt{u_c^2 + v_c^2}$.

For each bin $b \in \{\text{low}, \text{mid}, \text{high}\}$, we apply a band-pass mask to the power spectrum:

$$P_t^{(b)}(u, v) = P_t(u, v) \cdot \mathcal{M}_b(u, v), \tag{5}$$

and scale the original FFT to preserve phase:

$$\mathcal{F}_t^{(b)}(u, v) = \mathcal{F}_t(u, v) \cdot \sqrt{\frac{P_t^{(b)}(u, v)}{P_t(u, v) + \epsilon}}, \quad \epsilon = 10^{-10}. \tag{6}$$

The spatial representation is obtained via the inverse FFT:

$$M_t^{(b)}(h, w) = \left| \mathcal{F}^{-1}\{\mathcal{F}_t^{(b)}\} \right|. \tag{7}$$

We then normalize, so the frequency bands sum to the full magnitude:

$$\tilde{M}_t^{(b)}(h, w) = M_t^{(b)}(h, w) \cdot \frac{M_t(h, w)}{\sum_{b'} M_t^{(b')}(h, w) + \epsilon}. \tag{8}$$

This ensures $\sum_{b'} \tilde{M}_t^{(b')} = M_t$ at each pixel.

Visual Observations. The noise evolution visualization confirms that most noise change happens in the low-frequency components. These changes directly correspond to spatial changes in the generations throughout the optimization steps. This observation along with the fact that natural images have a 1/f power spectrum inspires our exploration of noise initializations with stronger low-frequency components (e.g. pink noise).

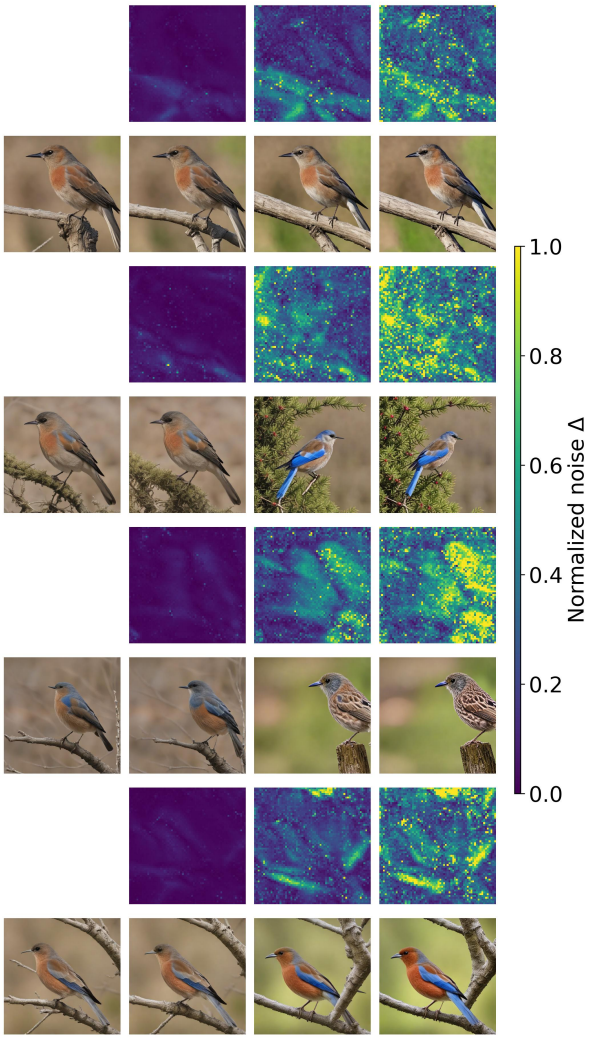


Figure 8. Noise evolution across optimization iterations for a set of four images. We show spatial heatmaps with the averaged L2 difference between the current noise latent and the initial white noise along with the corresponding generated image. Images were generated with SDXL-Turbo and the prompt: "A photo of a bird".

5.1. Pink Noise Example Generations

Higher α values (see main Eq. 6) generally lead to higher diversity scores. However, the image quality decreases with high noise exponents (see generations for $\alpha = 0.3$ and $\alpha = 0.5$ in Fig. 11 which have patchy artefacts). Note that we use CLIPScore as the only image quality reward during optimization for Tab. 5. However, additional rewards for image quality can easily be included in our pipeline.

In our experiments, we use a noise exponent of 0.2 (referred to as pink noise), which provides substantial gains in sample diversity and reduces the number of required iterations, while preserving image quality.

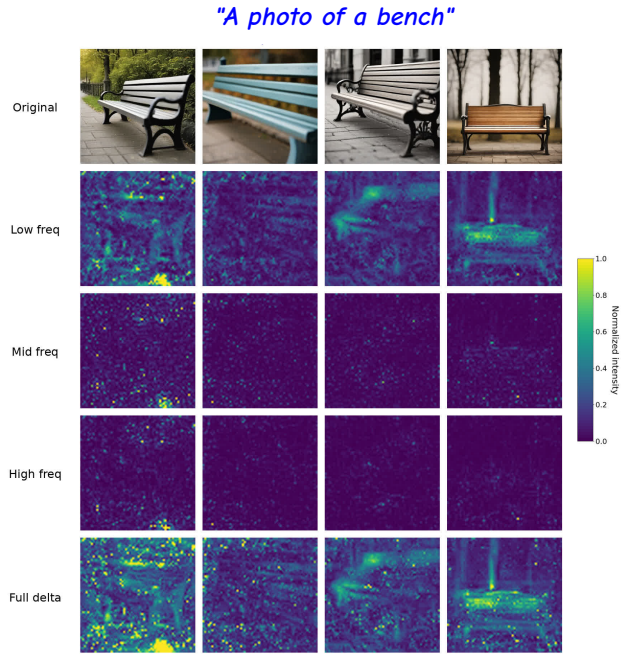


Figure 9. Example showing how the noise changes across optimization iterations in different frequency bands for SDXL-Turbo with white noise initialization and DINO diversity objective. We see that most of the change happens in the lowest third of the frequencies.

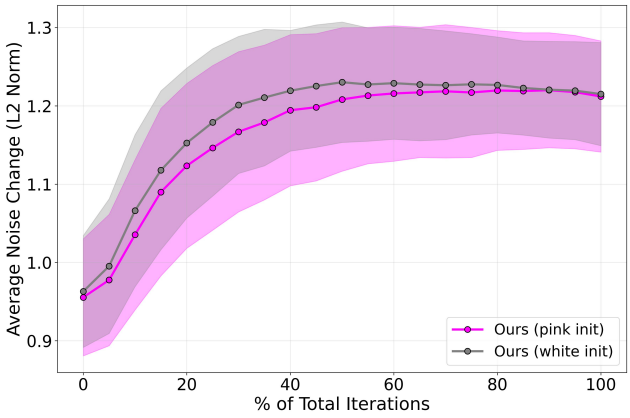


Figure 10. Noise change across iterations on raw noise signal measured as the L2 norm between subsequent iterations. White noise initialization results in slightly higher overall noise change across iterations than pink noise initialization.

6. User Study

We conduct a human user preference study to determine which methods produce more diverse outputs, similar to Parmar et al. [21]. We compare our method to baselines such as i.i.d. sampling and Parmar et al. [21], as well as across different target diversity objectives.

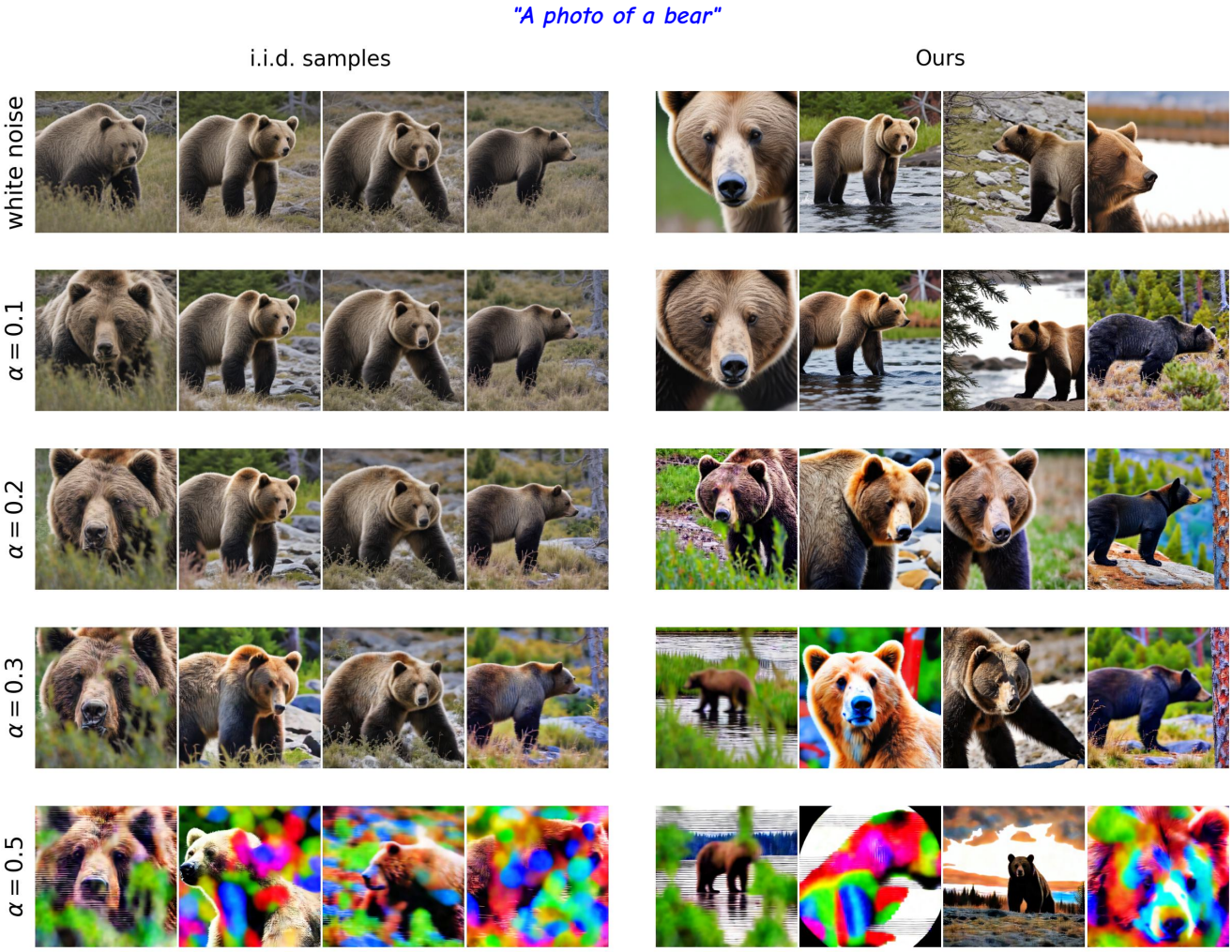


Figure 11. Effect of noise exponent values on image generation. Each row compares i.i.d. samples from initial noise (left) with our outputs (right) for different α values. Results were obtained with SDXL-Turbo and noise optimization using DINO diversity and CLIPScore.

Table 9. Human preference win rates from a user study for our method against i.i.d. sampling and Parmar et al. [21] for PixArt-α [2], SANA-Sprint-1.6B [3], and SDXL-Turbo [24].

Method	**Win % vs i.i.d.**	**Win % vs [21]**
PixArt-α [2]	90.00	77.50
SANA-Sprint-1.6B [3]	85.00	66.25
SDXL-Turbo [24]	88.75	91.25

During the study, we show participants a 2x2 grid of images generated from our method and a comparison. We ask the user to select "which grid of images has higher variety?". For each pairing, we collect 10 user preferences to determine a per prompt win rate. User data is anonymized and crowdsourced.

We run trials across all single-object prompts in the GenEval benchmark [9] (prompts 1 to 80). For reference, we also report diversity scores for this subset in Tab. 10. We count the number of wins across trials for each model to compute a final overall win percentage. In the results in Tab. 9, we observe that our method shows the highest win rate across all three models.

In addition, we compared our method across different diversity objectives (see main Fig. 6).

7. Failure Cases

Despite the effectiveness of our optimization approach, several failure modes can be observed. We visualize these in Fig. 12. When using DreamSim, the optimization sometimes produces blurry images as the method exploits perceptual distance which can remove high-frequency details (top row). Color histogram diversity tends to encourage plain backgrounds since uniform color regions efficiently

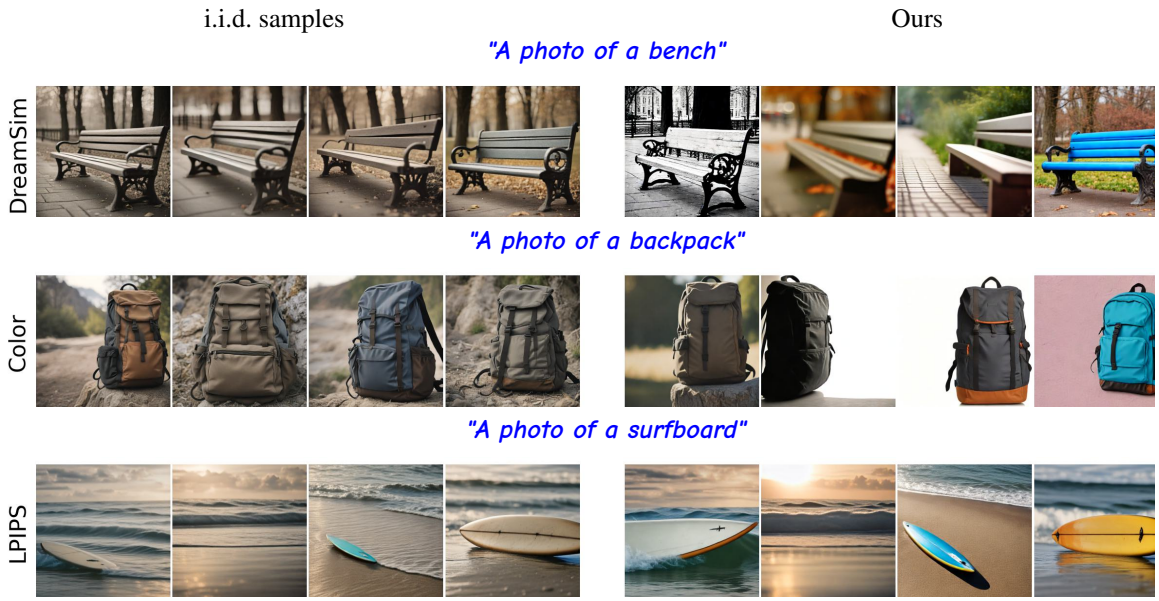


Figure 12. Failure cases of our method for different optimization objectives (SDXL-Turbo). Top row: Removing fine details through blurring one image increases perceptual distance without introducing meaningful diversity. Middle row: Overly simple compositions (e.g. plain backgrounds) lead to high color diversity scores as different solid colors maximize L2 color histogram distance effectively. Bottom row: LPIPS optimization fails to recover semantic content that is missing in the generation from the initial noise.

Table 10. Output diversity results on the single-object subset of GenEval for our proposed approach with the PixArt-α, SANA-Sprint-1.6B, and SDXL-Turbo models using white noise initialization. Output diversity is measured with averaged pairwise DINO, DreamSim, and LPIPS scores.

Method	DINO	DreamSim	LPIPS
PixArt-α [2]			
i.i.d.	$0.382_{\pm 0.093}$	$0.160_{\pm 0.078}$	$0.460_{\pm 0.126}$
Parmar et al. [21]	$0.520_{\pm 0.093}$	$0.227_{\pm 0.094}$	$0.563_{\pm 0.116}$
Ours	$0.731_{\pm 0.077}$	$0.370_{\pm 0.117}$	$0.691_{\pm 0.096}$
SANA-Sprint-1.6B [3]			
i.i.d.	$0.494_{\pm 0.091}$	$0.219_{\pm 0.081}$	$0.631_{\pm 0.070}$
Parmar et al. [21]	$0.695_{\pm 0.061}$	$0.363_{\pm 0.112}$	$0.733_{\pm 0.052}$
Ours	$0.752_{\pm 0.065}$	$0.485_{\pm 0.109}$	$0.795_{\pm 0.058}$
SDXL-Turbo [24]			
i.i.d.	$0.529_{\pm 0.077}$	$0.218_{\pm 0.089}$	$0.611_{\pm 0.058}$
Parmar et al. [21]	$0.667_{\pm 0.069}$	$0.320_{\pm 0.118}$	$0.661_{\pm 0.053}$
Ours	$0.808_{\pm 0.047}$	$0.450_{\pm 0.131}$	$0.768_{\pm 0.046}$

maximize histogram L2 distances. LPIPS diversity exhibits a critical limitation: it does not recover semantic content missing from the initial noise visualization (e.g., if a surfboard is not generated at first, it remains absent), as LPIPS diversifies existing perceptual features rather than introducing new semantic elements. This could be recovered with a larger weighting of image quality and prompt adherence rewards in the optimization process.

Figure 13. Impact of diversity objectives on the resulting noise optimization and image generations compared to i.i.d sampled noise initialization and the search method proposed by Parmar et al. [21]. Our approach results in more varied generations in terms of object pose, appearance, colors, and backgrounds (e.g. different horse breeds in different surroundings, and fire hydrants in different colors).

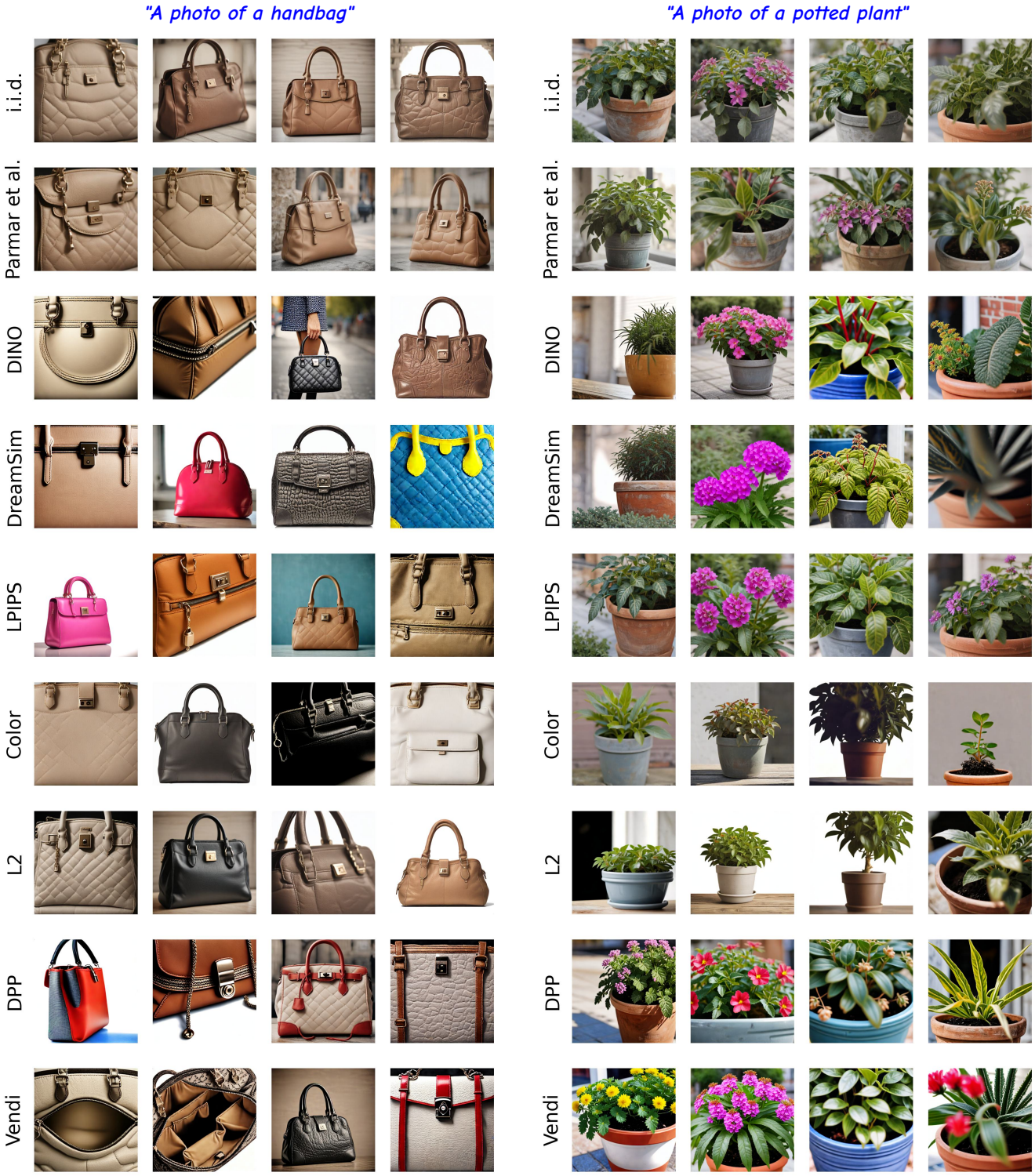


Figure 14. Impact of diversity objectives on the resulting noise optimization and image generations compared to i.i.d sampled noise initialization and the search method proposed by Parmar et al. [21]. The generated handbags and potted plants show larger variation in terms of handbag types and colors, and plant species.

References

[1] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 2

[2] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024. 2, 4, 11, 12

[3] Junsong Chen, Shuchen Xue, Yuyang Zhao, Jincheng Yu, Sayak Paul, Junyu Chen, Han Cai, Enze Xie, and Song Han. Sana-sprint: One-step diffusion with continuous-time consistency distillation. *arXiv preprint arXiv:2503.09641*, 2025. 2, 4, 11, 12

[4] Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi Jaakkola. Particle guidance: non-iid diverse sampling with diffusion models. *arXiv preprint arXiv:2310.13102*, 2023. 4, 9

[5] Mohamed Elfeki, Camille Couprie, Morgane Riviere, and Mohamed Elhoseiny. Gdpp: Learning diverse generations using determinantal point processes. In *ICML*, 2019. 3, 6

[6] Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. Reno: Enhancing one-step text-to-image models through reward-based noise optimization. *NeurIPS*, 2024. 1

[7] Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022. 1, 3

[8] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023. 1, 3

[9] Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *NeurIPS*, 2023. 2, 3, 4, 11

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017. 2

[11] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024. 2, 4, 6

[12] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *NeurIPS*, 2023. 4

[13] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *NeurIPS*, 2023. 2

[14] Alex Kulesza, Ben Taskar, et al. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 2012. 1

[15] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 2

[16] Black Forest Labs. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 2, 6, 8

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2

[18] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 3

[19] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. *NeurIPS Datasets and Benchmarks*, 2021. 2, 3

[20] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022. 2

[21] Gaurav Parmar, Or Patashnik, Daniil Ostashev, Kuan-Chieh Wang, Kfir Aberman, Srinivasa Narasimhan, and Jun-Yan Zhu. Scaling group inference for diverse and high-quality generation. *arXiv preprint arXiv:2508.15773*, 2025. 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 3

[23] Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann M Weber. Cads: Unleashing the diversity of diffusion models through condition-annealed sampling. *arXiv preprint arXiv:2310.17347*, 2023. 4, 5, 9

[24] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023. 2, 4, 8, 11, 12

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[26] Jaskirat Singh, Lindsey Li, Weijia Shi, Ranjay Krishna, Yejin Choi, Pang Wei Koh, Michael F Cohen, Stephen Gould, Liang Zheng, and Luke Zettlemoyer. Negative token merging: Image-based adversarial feature guidance. *arXiv preprint arXiv:2412.01339*, 2024. 4, 5, 9

[27] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *TPAMI*, 2008. 1, 3

[28] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. 2, 6

[29] 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. 1, 3