Guiding Token-Sparse Diffusion Models

Felix Krause Stefan Andreas Baumann Johannes Schusterbauer
Olga Grebenkova Ming Gui Vincent Tao Hu Björn Ommer
CompVis @ LMU Munich, Munich Center for Machine Learning (MCML)

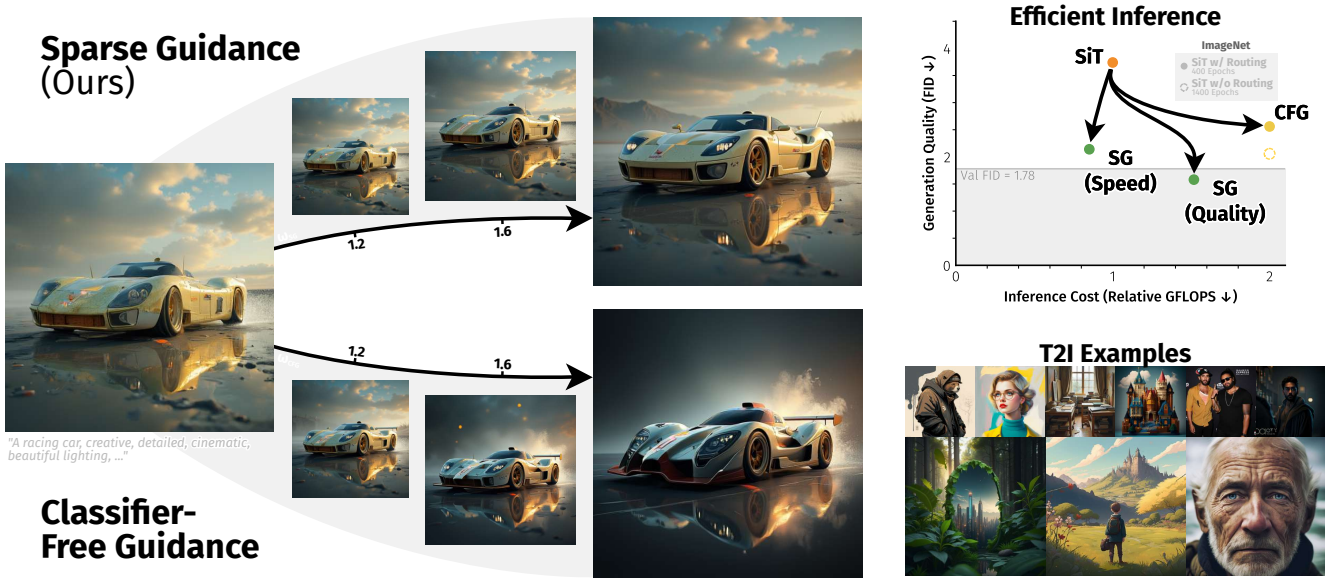

Figure 1. **Sparse Guidance provides effective, efficient, structure-preserving guidance for sparsely trained diffusion models.** (Left) Unlike Classifier-free Guidance, SG stays closer to the conditional prediction, yielding higher-variance, non-collapsed samples. (Right, top) On ImageNet-256, SG (Quality) attains an FID of *1.58* without any previously required dense finetuning while also increasing throughput, and SG (Speed) matches the baseline quality at substantially lower inference cost. (Right, bottom) Applied to our 2.5B text-to-image model, Sparse Guidance raises its HPSv3 [47] performance enough to surpass a range of larger models, which it could not achieve without SG.

Abstract

*Diffusion models deliver high quality in image synthesis but remain expensive during training and inference. Recent works have leveraged the inherent redundancy in visual content to make training more affordable by training only on a subset of visual information. While these methods were successful in providing cheaper and more effective training, sparsely trained diffusion models struggle in inference. This is due to their lacking response to Classifier-free Guidance (CFG) leading to underwhelming performance during inference. To overcome this, we propose Sparse Guidance (SG). Instead of using conditional dropout as a signal to guide diffusion models, SG uses token-level sparsity. As a result, SG preserves the high-variance of the conditional prediction better, achieving good quality and high variance outputs. Leveraging token-level sparsity at inference, SG improves fidelity at lower compute, achieving 1.58 FID on the commonly used ImageNet-256 benchmark with 25% fewer FLOPs, and yields up to 58% FLOP savings at matched baseline quality. To demonstrate the effectiveness of Sparse Guidance, we train a 2.5B text-to-image diffusion model using training time sparsity and leverage SG during inference. SG achieves improvements in composition and human preference score while increasing throughput at the same time. Project Page: **[TODO:]***

1. Introduction

In recent years, models developed by the machine learning community and industry have grown dramatically in size, thereby demanding massive computational resources [6,

Figure 2. **Classifier-free Guidance (CFG) provides limited benefits for token-sparse diffusion models.** While token-sparse training produces stronger conditional diffusion models than standard dense training, their practical impact has been constrained by poor compatibility with CFG, which limits inference quality and slows adoption in practice. **Sparse Guidance (SG) overcomes this limitation, restoring strong guidance gains for token-sparse models and enabling them to match or surpass the image quality of their dense baselines.**

27, 35, 70]. Diffusion models [23, 38, 62] have become a frequently used standard across modalities such as images [15, 35, 55] and video [5, 6, 70], despite being among the most compute-intensive approaches. Furthermore, Classifier-free Guidance (CFG) is commonly used for high generation quality. During CFG, an unconditional and a conditional prediction are combined, which typically doubles the inference costs of already very expensive diffusion models [22].

For the training of these models, methods like training-time sparsity [17, 32, 74] have shown improvements in efficiency as well as performance. These methods exploit the underlying redundancy of visual data and train a diffusion model only on a subset of available information at any given time. *Masking* replaces the discarded information with learnable parameters while *routing* aims to first withdraw and later reintroduce information. The reason the community has not adopted these approaches fully is a breakdown of inference capabilities: models trained with such training-time sparsity show unreliable and often weak performance during generation due to their unresponsiveness to CFG [32, 74, 77].

We propose *Sparse Guidance* (SG) as a direct remedy to the issue of costly inference and the practical usability of sparsely trained diffusion models at the same time. SG steers the generation process by leveraging a *capacity gap* induced by inference-time sparsity (i.e., a controlled difference between two predictions created by two distinct token-level sparsity rates). Unlike previous approaches [32, 60, 74], SG requires no additional finetuning to recover the model's capabilities under CFG while providing **higher quality** with **better throughput** as Sparse Guidance embraces the train-test gap of sparse training approaches instead of avoiding it. We validate SG on the commonly used ImageNet-256 benchmark, where SG achieves an FID of 1.58. Furthermore, we show predictable behavior and a smooth quality–throughput trade-off, where increasing inference-time sparsity reduces the number of processed tokens and lowers computational cost. Then we demonstrate that SG holds up at scale: we train a 2.5B text-to-image Diffusion Transformer using token routing [32] and, applying SG, find reliable improvements in image quality measured by human preference, alongside reduced FLOPs and increased inference throughput.

Our main contributions can be summarized as:

- We introduce *Sparse Guidance* (SG), a finetune-free, post-hoc scheduling mechanism for sparsely trained diffusion models. SG computes two predictions and applies token-level sparsity to them and then utilizes their capacity gap to steer the generation towards higher quality. As tokens are removed from the computational branch, the cost for inference shrinks naturally.
- Sparse Guidance delivers strong results without additional finetuning. SG achieves **FID 1.58** with **25%** fewer FLOPs, and up to **58%** savings at comparable quality to a dense SiT on the commonly used ImageNet-256 benchmark.
- To demonstrate the viability of this pipeline, we train a large scale text-to-image 2.5B Diffusion Transformer using token routing. We apply our proposed Sparse Guidance method which improves image quality measured by human preference score and naturally increases throughput during inference significantly by reducing the amount of processed information.

2. Related works

Diffusion and Flow Matching Models. Score-based diffusion models, such as DDPM [23] and its improved variants [48, 63–65], as well as Latent Diffusion Models [LDM, 55], have become the cornerstone of high-fidelity synthesis across images [53, 59], video [3, 24] and audio [26, 39, 49]. Complementarily, flow-matching methods [2, 38, 40, 43] recast generation as learning a continuous vector field within

an interpolant framework that unifies flow and diffusion, enabling efficient ODE-based sampling. Early diffusion frameworks relied on U-Net backbones [56], but recent work has shifted toward token-based transformers like DiT [51], which offer scalability at the cost of quadratic complexity in the number of tokens [74]. To mitigate this, caching schemes accelerate inference in both U-Nets [45] and DiTs [46], yet still process every token at each layer. In contrast, we utilize a test-time token-sparsity which allows us to reduce the number of processed tokens per layer.

Diffusion Guidance. Guidance has become a standard tool for improving the fidelity of diffusion model outputs. An auxiliary model or signal steers the generative process [12]. Currently, the most dominant approach is classifier-free guidance (CFG) [22], which combines the conditional and unconditional score to improve sample fidelity at the cost of diversity. Recent advances such as Autoguidance (AG) [30] use a smaller and less trained model to replace the previously used unconditional branch to achieve good guidance. Sadat et al. [57] apply perturbations to the timestep embeddings, causing intentional misalignment in noise removal to guide the generation process. Kaiser et al. [29] restrict the receptive field in convolution-based backbones for guidance. Beyond these classifier- and branch-based methods, attention-based schemes such as self-attention guidance [25] and perturbed-attention guidance [1] steer sampling by manipulating internal attention patterns. In contrast to previous methods, we propose to apply train-time sparsity augmentations to inference by using two token-sparsity rates (number of concurrently processed tokens) to create a capacity gap which we effectively use to steer the sampling process towards higher quality.

Token Sparsity. In parallel, efficiency-focused research has enabled models like the Transformer [69] to skip processing of less important tokens. Token masking has shown that the entire token set is not required for a diffusion model to approximate the data distribution [17, 74, 77]. The advantage in these methods is that training throughput is increased significantly, which reduces costs. As an alternative to masking, token routing reintroduces tokens instead of replacing them with learnable embeddings [32]. In the domain of diffusion models, such routing can preserve token information, providing better convergence speed while retaining the efficiency of similar masking methods. Relatedly, Mixture-of-Depths [54] employs a fixed top-k token selection per layer, which allows only k tokens to be processed by each layer, reducing computational cost. Beyond train-time masking and routing, test-time token merging and pruning in diffusion transformers reduce compute by compressing or dropping tokens while preserving visual quality. Furthermore, feature-caching approaches such as DeepCache and

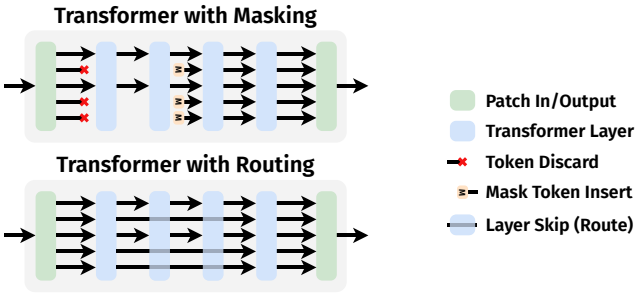

Figure 3. **Masking and Routing as two types of token-level sparsity.** Masking replaces tokens with learnable mask token [74] while routing preserves information by reintroducing tokens [32].

Learning-to-Cache accelerate diffusion U-Nets and transformers by reusing intermediate activations across timesteps or layers [45, 46]. Our method builds on train-time sparsity but introduces it to inference leveraging it as a guidance signal to improve visual quality.

3. Method

3.1. Preliminaries

Flow Matching. Flow Matching (FM) formulates generation as learning a continuous-time vector field that deterministically transports a simple prior distribution to the data distribution [2, 38, 40]. Concretely, let $z \sim \mathcal{N}(0, I)$ denote a latent sample from the prior and $x \sim p_{\text{data}}$ a corresponding data sample. We adopt the widely used standard straight (Gaussian) interpolation path [38]

$$x_t \;=\; (1-t)\,z \;+\; t\,x, \qquad t \in [0,1], \tag{1}$$

whose oracle velocity is constant along the path,

$$v^\star(x_t, t) \;=\; \frac{dx_t}{dt} \;=\; x - z. \tag{2}$$

A flow-matching model v_θ predicts $v^\star$, and sampling integrates the ODE $\frac{dx_t}{dt} = v_\theta(x_t, t)$ from $t = 0$ to $t = 1$ [38].

Classifer-free Guidance High-fidelity sampling often employs *Classifier-free Guidance* (CFG) to steer the conditional prediction away from a weaker (unconditional) branch. For brevity, we write $v_\theta(x_t, t, c)$ as $v_\theta(c)$ and retain only guidance-relevant terms. Given conditioning c and guidance scale $\omega \geq 1$, Classifier-free Guidance [22] is defined as:

$$v_\theta^{\text{CFG}}(c, \omega) \;=\; \omega\, v_\theta(c) \;+\; (1-\omega)\, v_\theta(\varnothing). \tag{3}$$

CFG doubles per-step compute for dense models. Our goal is to retain its benefits while *reducing* the compute increase under sparsity.

Token Sparsity. Let D_θ denote the denoiser network, composed of B sequential layers $L_0, \ldots, L_{B-1}$. Token sparsity reduces training cost by avoiding computation on the full set of tokens in every layer: *Masking* drops a fixed fraction γ of tokens and optionally replaces them with learnable embeddings, never re-inserting the original activations. We then define masking as follows:

$$D_\theta^{\mathbf{m}} = L_{B-1} \circ \cdots \circ \begin{Bmatrix} \text{mask}, & \tau_k \in \mathcal{T}_{\mathbf{m}} \\ L_k \circ \cdots \circ L_0, & \text{otherwise} \end{Bmatrix}, \quad (4)$$

where $\text{mask}(\tau_k) = e_{\text{mask}}$ replaces token τ_k with a fixed or learnable embedding that carries no instance-specific information, permanently removing the original activation from the forward path.

Routing selects a subset of tokens to process and re-inserts them later, keeping all tokens within the computational graph. This is then defined as:

$$D_\theta^{\mathbf{r}_{i\to j}} = L_{B-1} \circ \cdots \circ \begin{Bmatrix} \text{id}, & \tau_k \in \mathcal{T}_{\mathbf{r}_{i\to j}} \\ L_j \circ \cdots \circ L_i, & \text{otherwise} \end{Bmatrix} \circ \cdots \circ L_0, \quad (5)$$

where id denotes the identity mapping applied to routed tokens, ensuring they bypass intermediate layers while preserving their information for later re-insertion. Figure 3 demonstrates this visually.

3.2. Sparse Guidance (SG)

Using Training Augmentation as a Guidance Signal. Token-level sparsity has proven effective for accelerating *training* [17, 32, 77]. However, at *inference* time, models employing standard classifier-free guidance (CFG) frequently exhibit decreased response to the guidance signal or degraded fidelity unless subjected to dense finetuning (see Figure 2). We revisit sparsity not as a training-only device but as a *test-time control signal*. Formally, let $\gamma \in [0, 1)$ denote a sparsity rate that either masks tokens (replacement by a fixed/learnable embedding) or routes tokens (bypassing selected layers with identity and later reinsertion).

Controlling Capacity with Sparsity. Naively adapting a token-level sparsity $\gamma > 0$ during inference ($\omega = 1.0$) leads to deteriorated outputs Figure 4. As γ increases, the model's effective capacity shrinks, limiting its ability to realize the learned distribution and producing visually disturbing artifacts. To overcome this, we utilize the capacity-controlling sparsity knob γ during inference only in a guided setting. Guidance is most effective when a high-variance predictor pushes a lower-variance one toward outputs with even less variance (e.g., a specific conditioning) [22, 30, 34]. We find that token-level sparsity provides a direct knob for realizing this: increasing γ lowers effective capacity and *softens* the conditional distribution produced by $D_\theta(x_t, t, c; \gamma)$, while decreasing γ yields a sharper, higher-capacity predictor. We propose instantiating guidance by using a high-γ (weak) branch to steer a low-γ (strong) branch during sampling. The resulting capacity gap provides the guidance signal. In this view, γ is a single, continuous hyperparameter over distributional sharpness, turning train-time sparsity into a test-time *guidance primitive*.

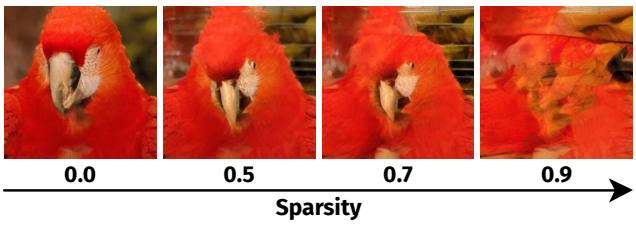

Figure 4. **Without Sparse Guidance, image quality and composition worsens consistently with increased token-sparsity ratios.**

Guidance Formulation. We evaluate the network D_θ under two test-time sparsity levels using the notation $D_\theta(x_t, t, c; \gamma)$ to indicate token sparsity γ. Further, we will define the two branches that are needed for a guided prediction as D_θ^{strong} and D_θ^{weak}, no matter what γ_{strong} or γ_{weak} is applied respectively.

$$\begin{aligned} D_\theta^{\text{strong}}(c) &:= D_\theta(x_t, t, c; \gamma_{\text{strong}}), \\ D_\theta^{\text{weak}}(c) &:= D_\theta(x_t, t, c; \gamma_{\text{weak}}), \\ & 0 \le \gamma_{\text{strong}} < \gamma_{\text{weak}} < 1. \end{aligned} \quad (6)$$

In contrast to CFG, both predictions are conditional. Consequently, the guidance signal is provided solely by the capacity gap induced by the difference in sparsity $\gamma_s \neq \gamma_{\text{weak}}$. Then we utilize the guidance formulation,

$$\begin{aligned} D_\theta^{\text{SG}}(c, \gamma_{\text{strong}}, \gamma_{\text{weak}}, \omega) = \omega\, D_\theta^{\text{strong}}(c) \\ + (1-\omega)\, D_\theta^{\text{weak}}(c) \end{aligned} \quad (7)$$

which uses the low-capacity, weak prediction $D_\theta^{\text{weak}}(c)$ to steer the high-capacity, strong prediction in the direction of $D_\theta^{\text{strong}}(c) - D_\theta^{\text{weak}}(c)$ with magnitude ω.

As SG makes no assumptions about the provided conditioning, it can be combined naturally with other existing guidance techniques. Applying the zero-condition $\varnothing$ to our weak branch leads to the combination of Classifier-free Guidance and Sparse Guidance (CFG + SG):

$$\begin{aligned} D_\theta^{\text{CFG+SG}}(c, \gamma_{\text{strong}}, \gamma_{\text{weak}}, \omega) = \omega\, D_\theta^{\text{strong}}(c) \\ + (1-\omega)\, D_\theta^{\text{weak}}(\varnothing). \end{aligned} \quad (8)$$

At test time, token subsets are sampled from binary masks $m \in \{0, 1\}^T$ with $m_k \sim \text{Bernoulli}(1-\gamma)$ for $\gamma \in \{\gamma_{\text{strong}}, \gamma_{\text{weak}}\}$.

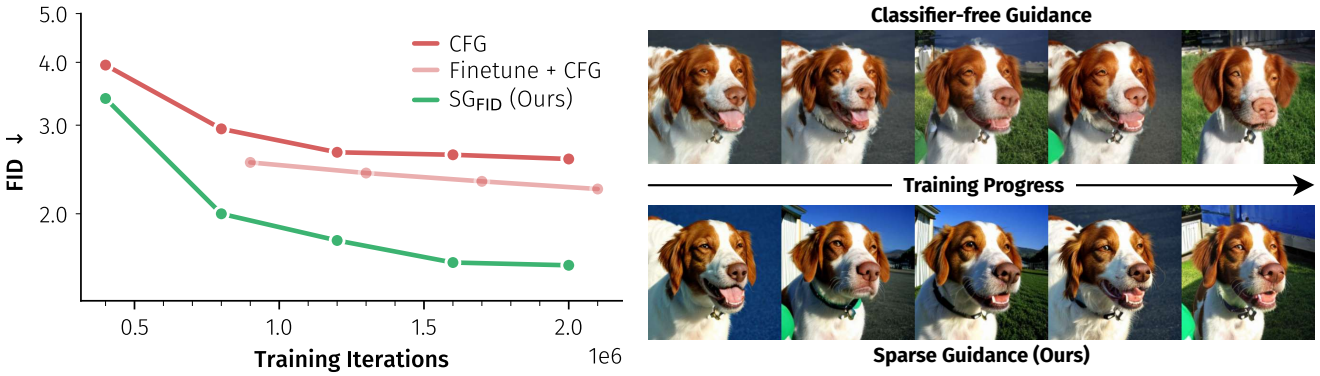

Figure 5. **Sparse Guidance improves both convergence and training-time sample quality for sparsely trained diffusion models. Left** FID over training iterations comparing CFG, CFG with dense finetuning, and Sparse Guidance (SG). where SG achieves the lowest FID using the best CFG scale ω for each method. **Right** Training-time sample progress using SG, showing that sparsely trained models already produce high-fidelity samples without an additional dense finetuning stage, enabling direct visual evaluation during training.

Hyperparameter Usage. Prior works applying sparsity during training often come with a variety of additional hyperparameters with their respective sparsity (or masking) rate being one of them [17, 32, 74]. Furthermore, several other guidance methods require affected layers to be handpicked for effective guidance [1, 28] while Sparse Guidance copies the train-time settings and applies them during inference leaving only γ as additional hyperparameter.

4. Experiments

We test our proposed Sparse Guidance method to leverage sparsely trained diffusion models during inference. To that end, we evaluate on class-conditional ImageNet-256 generation across model scales and compare to relevant guidance based baselines. Further, we provide evidence that Sparse Guidance and thereby indirectly sparse training methods as well, scale to billion parameter sized text-to-image models.

4.1. Experimental Setup

ImageNet Our experimental setup follows standard evaluation protocols, evaluating models in the class-conditional latent ImageNet-256^2 setting that the various methods [32, 51, 74] were developed for. To enable fair comparisons, we reproduce both a masking [MaskDiT, 74] and routing [TREAD, 32] model with the settings proposed in the respective works. We train using AdamW [41] at a learning rate of 1×10^{-4} at a batch size of 256 with default betas $(\beta_1, \beta_2) = (0.9, 0.999)$. We train both models as SiT-XL/2 [44, 51] models in the latent space of the Stable Diffusion [55] VAE. During inference, we sample using a simple euler sampler with 40 steps, unless noted otherwise. We evaluate samples using the standard established evaluation protocol, primarily relying on the Fréchet Inception Distance [FID, 21] for evaluation of generated sample quality. We use the standard implementation from ADM [12] and, unless noted otherwise, compute FID based on 50k random samples. In addition to FID, we also report sFID [13], Inception Score [IS, 58], and Precision and Recall [33] for our main results. We report further implementation details as well as comprehensive descriptions and details for all shown results achieved with Sparse Guidance in the Appendix.

Scaling up to Text2Image To test if Sparse Guidance works beyond ImageNet with small to medium-sized models, we train a 2.5B text-to-image diffusion transformer. We utilize the internVL3-2b [76] model as text encoder and apply a prompt prefix and insert a two layer transformer network between the Vision Language Model (VLM) and the Cross-Attention of our DiT as proposed by Ma et al. [42]. We use TREAD [32] as our training time sparsity and follow the proposed settings with a route from $L_2 \rightarrow L_{30}$ in a 34 layer network and 50% selection rate. We train our model on a recaptioned subset of COYO-700M [7] which sums up to 100M samples. We divide our training into two stages. In the first, we train on all 100M samples while in the second one, we filter our data according to aesthetics score and add synthetic data from JourneyDB [50] and FLUX-6M [16]. During inference, we use a 512×512 resolution with 50 euler sampling steps and apply `bfloat16`.

4.2. Sparse Guidance on ImageNet

Sparse Approaches We apply our Sparse Guidance to models trained using state-of-the-art sparse training methods. As dropping tokens is a shared process among token-sparse methods, the differentiating factor becomes the replacement dropped tokens. We decide on masking [74] and routing [32] as they embody extremes cases (discard information vs. reuse). SG shows improved generative quality for both of these approaches which demonstrates broad applicability.

Comparison against Guidance Methods. We evaluate *Sparse Guidance* against a broad suite of guidance techniques for sparsely trained generators. Across all settings, both SG_{FID} and SG_{FLOPS} consistently outperform alternative guidance methods on the same sparsely pretrained

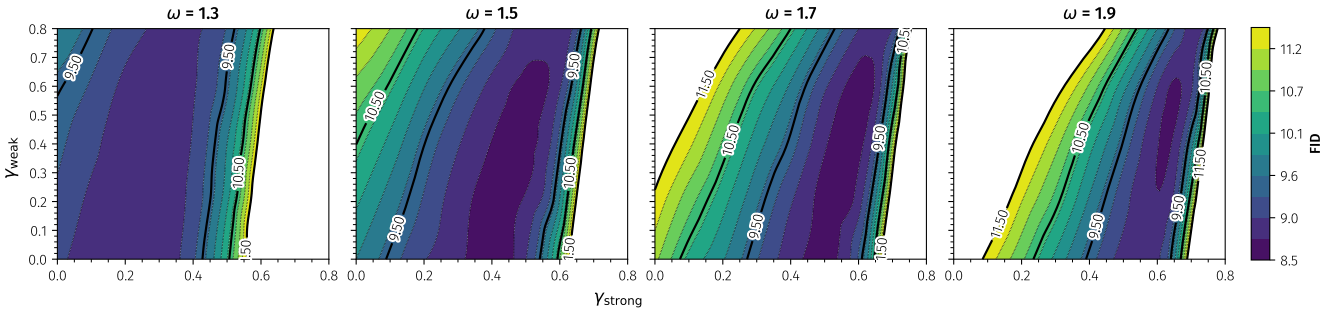

Figure 6. **Our method achieves lower FID robustly across different** ω by adaptation of γ_{strong} and γ_{weak}. We show the combination of AutoGuidance with Sparse Guidance and demonstrate how SG allows for fine grained control over the capacity gap between the D_θ^{strong} and D_θ^{weak} that drives guidance. Notably, the area of viable settings is broad and shifts under increasing ω towards higher γ_{strong} and γ_{weak}.

Guidance	Sparsity	#Epoch	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
CFG	masking	160	5.82	13.00	227.8	0.80	**0.45**
SG (Ours)	masking	160	**5.73**	**11.99**	**249.0**	**0.83**	0.42
CFG	routing	160	2.95	4.84	**233.3**	**0.82**	0.56
SG (Ours)	routing	160	**2.07**	**3.98**	223.4	0.80	**0.58**

Table 1. **SG improves upon CFG for diffusion models trained with masking and routing as their train-time sparsity.**

backbone. Notably, SG_{FID} achieves FID $= 1.58$ at 400 epochs, yielding a further 0.99 FID reduction over the next best competitor (CFG), indicating a substantive gain in perceptual quality. Beyond accuracy, SG reduces inference cost by enforcing sparsity at test time: SG_{FLOPS} attains lower GFLOPs than the no-guidance baseline while surpassing the baseline's quality with guidance. Under matched compute, SG also requires fewer operations than CFG, using **58%** fewer GFLOPs (SG_{FLOPS}). Furthermore, we compare to Independent Condition Guidance (ICG) [57] which introduces a guidance method without requiring training interventions, unlike CFG. We find that, SG achieves better performance than ICG which underlines our claim that Sparse Guidance minimizes the train-test gap by introducing test-time sparsity.

No Finetuning Requirements. Prior works observe irregular behavior when applying classifier-free guidance (CFG) to sparsity-augmented diffusion models have reported that an additional *dense* finetuning stage can partially restore CFG effectiveness [17, 32, 60, 74]. In Figure 5, we show that even after an extensive dense finetuning phase, CFG still fails to match the performance of our proposed Sparse Guidance method. Figure 5 mirrors these metrics with visual results on the right. Consequently, this supports our central claim that SG is *essential* to fully realize the generative capacity of sparsely trained diffusion models.

Method	#Epoch	FID↓	GFLOPS↓	ΔGFLOPS↓
SiT-XL/2 + routing	400	4.89	114.42	0 (baseline)
+CFG [22]	400	2.57	228.84	+114.42
+AG [30]	400	2.95	228.84	+114.42
+ICG [57]	400	2.81	228.84	+114.42
+SG_{FLOPS} (Ours)	400	2.14	**97.67**	**-16.75**
+SG_{FID} (Ours)	400	**1.58**	173.16	+58.74

Table 2. **SG outperforms other guidance methods by significant margins** in FID and GFLOPS. ΔGFLOPS is computed relative to the unguided baseline.

State-of-the-Art Comparison. Finally, we also compare with state-of-the-art diffusion models in Table 3. Using our high-quality configuration SG_{FID} , we achieve an FID of 1.58, outperforming a multitude of baselines while simultaneously offering a significant 24.6% reduction in inference cost compared to a dense guided SiT baselines (173.16 vs 228.84 GFLOPS). Aside from FID, SG_{FID} also provides larger recall [33], indicating higher variance in sampled images.

4.3. Effect of Sparsity

At inference, we impose distinct sparsity rates on the two branches: γ_{strong} on D_θ^{strong} and γ_{weak} on D_θ^{weak}. To study the behavior of these hyperparameters and their interaction with the guidance scale ω, we evaluate the triplet $(\gamma_{\text{strong}}, \gamma_{\text{weak}}, \omega)$ across a range of combinations. For greater coverage of the configuration space, we report FID@5k, enabling a more exhaustive analysis than standard evaluation settings.

Method	#Epoch	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
DiT-XL/2 [51]	1400	2.27	4.60	278.24	**0.83**	0.57
SD-DiT-XL/2 [77]	480	3.23	–	–	–	–
FasterDiT-XL/2 [71]	400	2.03	4.63	264.00	0.81	0.60
MaskDiT-XL/2 [74]	1600	2.28	5.67	276.56	0.80	0.61
MDT-XL/2 [17]	1300	1.79	4.57	283.01	0.81	0.61
SiT-XL/2 [43]	1400	2.06	4.50	270.30	0.82	0.59
SiT-XL/2 + REPA [72]	800	1.80	4.50	**284.00**	0.81	0.61
SiT-XL/2 + routing [32]*	400	2.57	4.99	275.26	0.82	0.57
+ SG_{FID} (Ours)	400	**1.58**	**4.45**	249.70	0.80	**0.63**

Table 3. **SG achieves 1.58 FID on the ImageNet-256 benchmark.** * denotes our reproduced experiments.

Guidance scale and sparsity. Figures 6 and 8 vary the guidance scale ω alongside the sparsity controls $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$. Across $\omega \in \{1.3, 1.5, 1.7, 1.9\}$ the optimal FID remains essentially unchanged, yet larger ω consistently tolerates higher total sparsity induced by $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$. Consequently, jointly increasing ω and $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$ improves efficiency while maintaining image quality. Figure 6 visualizes this with FID heatmaps whose color range is clipped to highlight the trend. The $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$ valley shifts and steepens as ω increases. The optimum becomes more localized and flattens less while permitting higher sparsity. Intuitively, larger ω pairs well with higher inference-time sparsity because sparsity degrades the generated signal. This pushes samples farther from the target image manifold while stronger guidance scale ω counteracts this drift.

Routing vs. Masking. Routing withholds tokens temporarily and reinserts them unchanged, preserving instance-specific information and stabilizing guidance. Accordingly, the $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$ landscape is broader, supports higher total sparsity, and is less sensitive to hyperparameters. Masking entails irreversible token deletion but even in this regime SG remains effective. As expected, the response surface over $(\gamma_{\text{strong}}, \gamma_{\text{weak}})$ is narrower than that found in routing but a clear corridor achieves improved FID (see Figure 8). This demonstrates that even sparsities which intuitively do not align with the iterative refinement goal of diffusion, can still be used to effectively guide the model towards better quality using our proposed Sparse Guidance method.

Compounding Gains with AutoGuidance. We further evaluate compatibility with external guidance by incorporating undertrained auxiliary models, following Karras et al. [30], within our Sparse Guidance (SG) framework. A central limitation of *AutoGuidance* is the requirement for an additional training run with dense checkpointing: only a narrow window of auxiliary checkpoints yields high-quality results, and Karras et al. [30] recommend dedicating $\frac{1}{16}$ of the total training iterations to the auxiliary model. This design is inherently inflexible, as the checkpoint cadence must be selected *a priori*. In contrast, SG markedly relaxes these

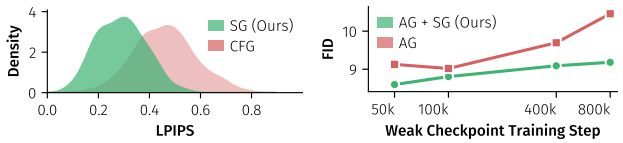

Figure 7. **(Left) SG demonstrates smaller LPIPS between the output with guidance and the conditional prediction. (Right) SG allows for better usage of other, less flexible guidance methods, like AutoGuidance** by offering the capability to adjust network capacities without training for fine-grained capacity gaps.

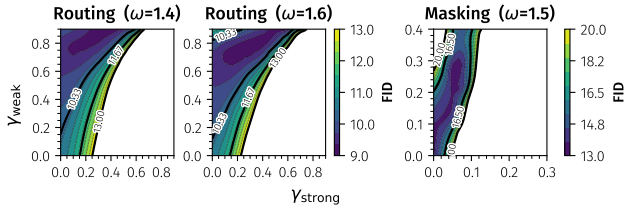

Figure 8. **Sparse Guidance provides qualitative improvements on routing and masking models** and demonstrates well behaved trade-off between $(\gamma_{\text{strong}}, \gamma_{\text{weak}}$ and $\omega)$ where larger ω allows for higher rates of sparsity and therefore also higher throughput.

constraints. Instead of relying on a precise reference checkpoint, (near-) optimal auxiliary models can be recovered from a broad range of training steps by tuning the sparsity controls γ_{strong} and γ_{weak}. As shown in Figure 7, we evaluate auxiliary checkpoints at 50k, 100k, 400k, and 800k steps—corresponding to 2.5%, 5%, 20%, and 40% of the total training iterations of v_0. For later checkpoints (800k and 400k), the best FID is achieved with $\gamma_{\text{strong}} = 0.0$. As we move to earlier checkpoints, the optimal γ_{strong} for v_0 increases to preserve the relative gap between the v_0 and v_1 output distributions. Overall, SG broadens the set of usable auxiliary checkpoints and compensates for their suboptimality through sparsity adaptation, delivering a favorable balance between FID and inference efficiency without committing to rigid checkpoint schedules.

4.4. Sparse Guidance in large scale T2I models

To provide insights into a more complex task at scale, we train a 2.5B diffusion transformer with routing sparsity according to Krause et al. [32]. We evaluate our model using standard CFG and our proposed Sparse Guidance on common benchmarks like GenEval [18] and HPSv3 [47]. Instead of FID, we utilize HPSv3 as our metric of choice to determine sparsity rates γ_{strong} and γ_{weak}. For this we use 250 synthetically generated prompts and the mean score over these. Phenomena previously reported at small scale on ImageNet-256 also persist in our billion-parameter text-to-image setting: even without any guidance. TR-DiT-2.5B's conditional branch exhibits clear, prompt- and layout-aware structure, consistent with the analysis of Krause et al. [32]. Furthermore, we confirm that Classifier-free Guidance (CFG) pulls the conditional predictor toward more stereotypical solutions. This aligns with the elevated *Recall* we measure for SG in Table 3 and the qualitative trend in Figure 9.

Visual Variance. Aside from oversaturation, CFG is known for variance-collapsing properties due to the fact that one extrapolates away from the unconditional signal in the direction of the conditional signal. While this is effective in overall image-prompt alignment, CFG can quickly produce similar looking images, especially with rare permutations

Model	Rank ↓	Overall ↑	Characters	Arts	Design	Architecture	Animals	Natural Scenery	Transportation	Products	Others	Plants	Food	Science
Kolors [31]	1	10.55	11.79	10.47	9.87	10.82	10.60	9.89	10.68	10.93	10.50	10.63	11.06	9.51
Flux-dev [35]	2	10.43	11.70	10.32	9.39	10.93	10.38	10.01	10.84	11.24	10.21	10.38	11.24	9.16
Playgroundv2.5 [36]	3	10.27	11.07	9.84	9.64	10.45	10.38	9.94	10.51	10.62	10.15	10.62	10.84	9.39
Infinity [20]	4	10.26	11.17	9.95	9.43	10.36	9.27	10.11	10.36	10.59	10.08	10.30	10.59	9.62
TR-DiT-2.5B + SG (Ours)	5	9.87	11.32	9.45	9.15	10.21	9.82	9.01	10.39	10.41	9.57	9.81	10.82	8.42
CogView4 [75]	6	9.61	10.72	9.86	9.33	9.88	9.16	9.45	9.69	9.86	9.45	9.49	10.16	8.97
PixArt-Σ [8]	7	9.37	10.08	9.07	8.41	9.83	8.86	8.87	9.44	9.57	9.52	9.73	10.35	8.58
Gemini 2.0 Flash [19]	8	9.21	9.98	8.44	7.64	10.11	9.42	9.01	9.74	9.64	9.55	10.16	7.61	9.23
TR-DiT-2.5B + CFG	9	9.21	10.54	9.33	9.15	9.34	9.41	8.44	9.36	9.51	8.57	9.34	10.42	8.60
Stable Diffusion XL [52]	10	8.20	8.67	7.63	7.53	8.57	8.18	7.76	8.65	8.85	8.32	8.43	8.78	7.29
HunyuanDiT [37]	11	8.19	7.96	8.11	8.28	8.71	7.24	7.86	8.33	8.55	8.28	8.31	8.48	8.20
TR-DiT-2.5B (Unguided)	12	7.76	8.49	8.04	8.33	7.97	6.63	7.77	7.40	7.38	7.02	8.02	8.06	8.01
Stable Diffusion 3 Medium [15]	13	5.31	6.70	5.98	5.15	5.25	4.09	5.24	4.25	5.71	5.84	6.01	5.71	4.58
Stable Diffusion 2 [66]	14	-0.24	-0.34	-0.56	-1.35	-0.24	-0.54	-0.32	1.00	1.11	-0.01	-0.38	-0.38	-0.84

Table 4. **HPSv3 scores for our sparsely trained TR-DiT-2.5B. SG improves over CFG in all categories** and enables our model to beat three additional models (`Gemini 2.0 Flash`, `PixArt-Σ` and `CogView4`). More precisely, our method improves sample quality by 27% over the unguided model and 7% over the model using CFG while increasing throughput from 0.32 to 0.49 images/s on an H200 GPU.

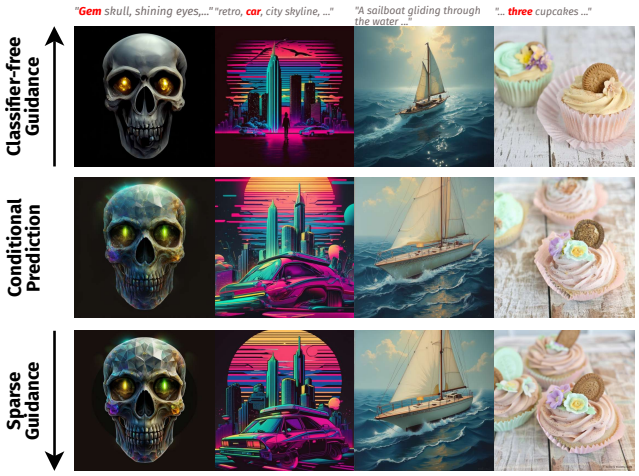

Figure 9. **Selected examples: Sparse Guidance keeps more of the structure of the conditional prediction** leading to higher variance in sample distribution while staying truthful to the prompt.

on otherwise common objects (see Figure 9). Since Sparse Guidance utilizes token sparsity as a driving force for guidance, instead of the text conditioning, we find that it retains the high-variance, creative expressivity of the conditional prediction better. This is shown in Figure 1 and Figure 9.

Performance Comparison. We evaluate TR-DiT-2.5B on the GenEval benchmark [18], which assesses compositional text–image alignment across six categories: *single object*, *two objects*, *counting*, *colors*, *relative position*, and *color attribution*. GenEval uses off-the-shelf detectors and classifiers to verify prompt satisfaction. With a standard Classifier-free Guidance (CFG) setting, TR-DiT-2.5B attains an overall score of 0.61. Incorporating our proposed SG method yields a score of 0.62, indicating a consistent improvement attributable to SG (see Table 5). SG improves performance in every category, evidencing a robust guidance signal for compositional grounding. Notably, on GenEval's everyday-object prompts, where CFG already excels via

Model	Overall ↑	Single object	Two object	Counting	Colors	Position	Color attribution
Stable Diffusion v1.5 [55]	0.43	0.97	0.38	0.35	0.76	0.04	0.06
Stable Diffusion v2.1 [66]	0.50	0.98	0.51	0.44	0.85	0.07	0.17
Stable Diffusion XL [52]	0.55	0.98	0.74	0.39	0.85	0.15	0.23
PixArt-alpha [9]	0.48	0.98	0.50	0.44	0.80	0.08	0.07
Flux.1-dev [35]	0.66	0.98	0.79	0.73	0.77	0.22	0.45
DALL-E 3 [4]	0.67	0.96	0.87	0.47	0.83	0.43	0.45
CogView4 [75]	0.73	0.99	0.86	0.66	0.79	0.48	0.58
Stable Diffusion 3 Medium [15]	0.74	0.99	0.94	0.72	0.89	0.33	0.60
Janus-Pro-7B [10]	**0.80**	0.99	0.89	0.59	0.90	**0.79**	0.66
TR-DiT-2.5B (Unguided)	0.48	0.93	0.50	0.36	0.77	0.13	0.20
TR-DiT-2.5B + CFG	0.61	0.98	0.73	0.55	0.86	0.19	0.36
TR-DiT-2.5B + SG	0.62	0.99	0.73	0.55	0.87	0.20	0.39

Table 5. **GenEval scores for our sparsely trained TR-DiT-2.5B.** SG shows consistent improvements over CFG.

variance-collapsing, prompt-faithful generation, SG still yields additional gains. We also show that our method can not only generate more correct images, as shown in GenEval, but also more visually appealing ones. In Table 4 we show HPSv3 scores taken from Ma et al. [47] and find that the addition of SG improves our model from matching `Gemini 2.0 Flash` to beating `CogView4` in overall score. In other words, SG allows our model to beat three additional models that it was previously not able to outperform.

5. Conclusion

Sparse training approaches for diffusion models have shown large improvements in recent years, but lacked adaption by the community as their performance and behavior during inference was unpredictable and weak. To overcome this, we propose Sparse Guidance (SG) which erases this issue and provides additional benefits like a higher variance in sampled outputs as well as fine-grained control over the capacity gap driving guidance. With SG we achieve an **FID of 1.58** while reducing FLOPs by **25%**, and can push to a **58%** FLOPs reduction at performance on par with the dense SiT baseline. Then, we scale sparse training to 2.5B for a text-to-image task and find SG holds up at scale, improving human preference score and increasing throughput. We hope that our work encourages the community to experiment with token-sparse diffusion models as this would lead to massive savings in cost, compute and CO_2.

Acknowledgments

We would like to thank Shih-Ying Yeh, Rami Seid, David Glukhov, and Swayam Bhanded for the insightful discussions. This project has been supported by the project "GeniusRobot" (01IS24083) funded by the Federal Ministry of Research, Technology and Space (BMFTR), the Horizon Europe project ELLIOT (101214398), the project "NXT GEN AI METHODS - Generative Methoden für Perzeption, Prädiktion und Planung" of the Federal Ministry for Economic Affairs and Energy (BMWE), and the bidt project KLIMA-MEMES. The authors gratefully acknowledge the Gauss Center for Supercomputing for providing compute through the NIC on JUWELS/JUPITER at JSC and the HPC resources supplied by the NHR @ FAU Erlangen. Further, we would like to thank Owen Vincent for continuous technical support.

References

[1] Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *European Conference on Computer Vision*, pages 1–17. Springer, 2024. 3, 5

[2] Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023. 2, 3

[3] Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 2

[4] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023. 8

[5] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

[6] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1, 2

[7] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 5, 1

[8] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation, 2024. 8

[9] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *ICLR*, 2024. 8

[10] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025. 8

[11] Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Forty-first International Conference on Machine Learning*, 2024. 1

[12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 2021. 3, 5

[13] Xin Ding, Yongwei Wang, Zuheng Xu, William J Welch, and Z Jane Wang. Continuous conditional generative adversarial networks: Novel empirical losses and label input mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8143–8158, 2022. 5

[14] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[15] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. *ICML*, 2024. 2, 8

[16] Rongyao Fang, Aldrich Yu, Chengqi Duan, Linjiang Huang, Shuai Bai, Yuxuan Cai, Kun Wang, Si Liu, Xihui Liu, and Hongsheng Li. Flux-reason-6m & prism-bench: A million-scale text-to-image reasoning dataset and comprehensive benchmark. *arXiv preprint arXiv:2509.09680*, 2025. 5, 1

[17] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Masked diffusion transformer is a strong image synthesizer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23164–23173, 2023. 2, 3, 4, 5, 6

[18] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023. 7, 8, 2

[19] Google DeepMind. Gemini 2.0 flash: Model card. https://modelcards.withgoogle.com/assets/documents/gemini-2-flash.pdf, 2025. 8

[20] Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis, 2024. 8

[21] 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. In *NeurIPS*, 2017. 5

[22] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop*, 2021. 2, 3, 4, 6

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[24] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *arXiv*, 2022. 2

[25] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7462–7471, 2023. 3

[26] Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. In *International Conference on Machine Learning*, pages 13916–13932. PMLR, 2023. 2

[27] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 2

[28] Junha Hyung, Kinam Kim, Susung Hong, Min-Jung Kim, and Jaegul Choo. Spatiotemporal skip guidance for enhanced video diffusion sampling. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11006–11015, 2025. 5

[29] Tim Kaiser, Nikolas Adaloglou, and Markus Kollmann. The unreasonable effectiveness of guidance for diffusion models. *arXiv preprint arXiv:2411.10257*, 2024. 3

[30] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *Advances in Neural Information Processing Systems*, 37:52996–53021, 2024. 3, 4, 6, 7

[31] Kolors Team. Kolors: Effective training of diffusion model for photorealistic text-to-image synthesis. https://huggingface.co/Kwai-Kolors/Kolors, 2024. Technical report and model card. 8

[32] Felix Krause, Timy Phan, Ming Gui, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Tread: Token routing for efficient architecture-agnostic diffusion training. *arXiv preprint arXiv:2501.04765*, 2025. 2, 3, 4, 5, 6, 7, 1

[33] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019. 5, 6

[34] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *NeurIPS*, 2024. 4

[35] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 2, 8

[36] Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation. *arXiv preprint arXiv:2402.17245*, 2024. 8

[37] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, Dayou Chen, Jiajun He, Jiahao Li, Wenyue Li, Chen Zhang, Rongwei Quan, Jianxiang Lu, Jiabin Huang, Xiaoyan Yuan, Xiaoxiao Zheng, Yixuan Li, Jihong Zhang, Chao Zhang, Meng Chen, Jie Liu, Zheng Fang, Weiyan Wang, Jinbao Xue, Yangyu Tao, Jianchen Zhu, Kai Liu, Sihuan Lin, Yifu Sun, Yun Li, Dongdong Wang, Mingtao Chen, Zhichao Hu, Xiao Xiao, Yan Chen, Yuhong Liu, Wei Liu, Di Wang, Yong Yang, Jie Jiang, and Qinglin Lu. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024. 8

[38] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023. 2, 3

[39] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. In *ICML*, 2023. 2

[40] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 2, 3

[41] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 5, 1

[42] Bingqi Ma, Zhuofan Zong, Guanglu Song, Hongsheng Li, and Yu Liu. Exploring the role of large language models in prompt encoding for diffusion models. *arXiv preprint arXiv:2406.11831*, 2024. 5, 1

[43] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024. 2, 6

[44] Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers, 2024. 5

[45] Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free, 2023. 3

[46] Xinyin Ma, Gongfan Fang, Michael Bi Mi, and Xinchao Wang. Learning-to-cache: Accelerating diffusion transformer via layer caching, 2024. 3

[47] Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15086–15095, 2025. 1, 7, 8, 2

[48] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 2

[49] Javier Nistal, Marco Pasini, Cyran Aouameur, Maarten Grachten, and Stefan Lattner. Diff-a-riff: Musical accompaniment co-creation via latent diffusion models. *arXiv preprint arXiv:2406.08384*, 2024. 2

[50] Junting Pan, Keqiang Sun, Yuying Ge, Hao Li, Haodong Duan, Xiaoshi Wu, Renrui Zhang, Aojun Zhou, Zipeng Qin, Yi Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Journeydb: A benchmark for generative image understanding, 2023. 5, 1

[51] William Peebles and Saining Xie. Scalable diffusion models with transformers. *ICCV*, 2023. 3, 5, 6, 1

[52] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. 8

[53] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2

[54] David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-depths: Dynamically allocating compute in transformer-based language models. *arXiv preprint arXiv:2404.02258*, 2024. 3

[55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 5, 8, 1

[56] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 2015. 3

[57] Seyedmorteza Sadat, Manuel Kansy, Otmar Hilliges, and Romann M Weber. No training, no problem: Rethinking classifier-free guidance for diffusion models. *arXiv preprint arXiv:2407.02687*, 2024. 3, 6

[58] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016. 5

[59] Johannes Schusterbauer, Ming Gui, Pingchuan Ma, Nick Stracke, Stefan A. Baumann, Vincent Tao Hu, and Björn Ommer. Boosting latent diffusion with flow matching. In *ECCV*, 2024. 2

[60] Vikash Sehwag, Xianghao Kong, Jingtao Li, Michael Spranger, and Lingjuan Lyu. Stretching each dollar: Diffusion training from scratch on a micro-budget. *arXiv preprint arXiv:2407.15811*, 2024. 2, 6

[61] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 1

[62] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 2

[63] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 2

[64] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[65] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[66] Stability AI. Stable diffusion 2.0 release. https://stability.ai/news/stable-diffusion-v2-release, 2022. 8

[67] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 1

[68] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1

[69] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 3, 1

[70] Veo-Team, :, Agrim Gupta, Ali Razavi, Andeep Toor, Ankush Gupta, Dumitru Erhan, Eleni Shaw, Eric Lau, Frank Belletti, Gabe Barth-Maron, Gregory Shaw, Hakan Erdogan, Hakim Sidahmed, Henna Nandwani, Hernan Moraldo, Hyunjik Kim, Irina Blok, Jeff Donahue, José Lezama, Kory Mathewson, Kurtis David, Matthieu Kim Lorrain, Marc van Zee, Medhini Narasimhan, Miaosen Wang, Mohammad Babaeizadeh, Nelly Papalampidi, Nick Pezzotti, Nilpa Jha, Parker Barnes, Pieter-Jan Kindermans, Rachel Hornung, Ruben Villegas, Ryan Poplin, Salah Zaiem, Sander Dieleman, Sayna Ebrahimi, Scott Wisdom, Serena Zhang, Shlomi Fruchter, Signe Nørly, Weizhe Hua, Xinchen Yan, Yuqing Du, and Yutian Chen. Veo 2. 2024. 2

[71] Jingfeng Yao, Cheng Wang, Wenyu Liu, and Xinggang Wang. Fasterdit: Towards faster diffusion transformers training without architecture modification. *Advances in Neural Information Processing Systems*, 37:56166–56189, 2024. 6

[72] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024. 6

[73] Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in neural information processing systems*, 32, 2019. 1

[74] Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *TMLR*, 2024. 2, 3, 5, 6, 1

[75] ZhipuAI. Cogview4-6b. https://huggingface.co/zai-org/CogView4-6B, 2025. 8

[76] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 5, 1

[77] Rui Zhu, Yingwei Pan, Yehao Li, Ting Yao, Zhenglong Sun, Tao Mei, and Chang Wen Chen. Sd-dit: Unleashing the power of self-supervised discrimination in diffusion transformer. In *CVPR*, pages 8435–8445, 2024. 2, 3, 4, 6

A. Implementation Details

A.1. Training Details for T2I

Architecture We implement our transformer models [14, 69] largely following the Llama architecture [68]. In particular, we apply pre-normalization via RMSNorm [73], exclude bias parameters from all linear transformations, and employ rotary positional embeddings [67] in an axial configuration following the approach of Crowson et al. [11]. The feedforward network (FFN) design mirrors that of Llama, utilizing the SwiGLU activation [61] and an expansion ratio of $\frac{8}{3}$.

Model We train a modern T2I diffusion transformer with 2.5B parameters. To apply TREAD [32], we mask tokens and positional indices simultaneously and reintroduce them at layer 30. We use Internvl3-2B [76] as the text encoder. In addition, we incorporate insights from Ma et al. [42], specifically employing two TransformerLayers after the frozen VLM and using a general system prompt as a prefix to our captions: "`Describe the image by detailing the color, shape, size, texture, quantity, text, and spatial relationships of the objects.`". For more details on the model refer to Table A1.

Data We use InternVL3-2B [76] to recaption a 100M-sample subset of COYO-700M [7], producing four captions per image. First, we generate a highly detailed description of the image and then progressively distill it into three additional levels: multi-sentence descriptions, single-sentence descriptions, and finally keyword-level summaries. For the last three, we use the language capacity of the VLM exclusively to cut down on cost. After a first training stage, we filter the COYO subset by aesthetics score (>5) and add synthetic data from JourneyDB [50] and Flux-6M [16].

Hyperparameter	TR-DiT-2.5B
Optimizer	
Batch size	3,072
Optimizer	AdamW
Learning rate	5×10^{-5}
(β_1, β_2)	(0.9, 0.95)
Architecture	
Embedding dim	2,048
Attention heads	16
Transformer layers	34
TREAD settings	
Route	$\mathbf{r}_{2\to30}$
Selection ratio	0.5

Table A1. Hyperparameter setup for our TR-DiT-2.5B model and the TREAD routing schedule.

A.2. Hyperparameters for ImageNet

Unless stated otherwise we inherit the DiT [51] setting: AdamW [41], a fixed learning rate of 10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, `bf16` precision, and latent-space training with the `stabilityai/sd-vae-ft-ema` VAE [55]. When we finetune LR is dropped to 10^{-5}. For routing and masking specific parameters refer to Table A2.

Hyperparameter	Routing	Masking
Optimizer		
Batch size	256	256
Optimizer	AdamW	AdamW
Learning rate	1×10^{-4}	1×10^{-4}
(β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)
Finetune		
Batch size	256	256
Learning rate	1×10^{-5}	1×10^{-5}
Architecture		
Embedding dim	1,152	1,152
Attention heads	16	16
Transformer layers	28	28
TREAD settings		
Route	$\mathbf{r}_{2\to24}$	–
Selection ratio	0.5	–
MaskDiT settings		
D^{dec}Embedding dim	–	512
D^{dec}Attention heads	–	16
D^{dec}Transformer layers	–	8
Selection ratio	–	0.5

Table A2. Hyperparameter setup for the XL/2 backbones with additional information for routing [32] and masking [74] methods. D^{dec} refers to the decoder head placed upon the normal DiT-XL/2. $\mathbf{r}_{2\to24}$ refers to the route from layer 2 to layer 24.

B. Experiment Details

B.1. Sparse Guidance in ImageNet

SG$_{\text{FLOPS}}$ from Section 4.2 is obtained using the same checkpoint for the high capacity and low capacity model. Both are conditional and the distribution discrepancy is created solely via different routing rates. We find $\gamma_{\text{strong}} = 0.5, \gamma_{\text{weak}} = 0.9$ to achieve good FID while substantially decreasing FLOPS.

SG$_{\text{FID}}$ (see Section 4.2, Table 3) is obtained through the usage of an early checkpoint of the same model training run. More specifically, we utilize a checkpoint with 50k training iterations. Furthermore, we apply cosine decay from 0.6 to 0.0 on the auxiliary model and the inverse on the main model. This aligns with the findings from Figure 7 where $\gamma_{\text{strong}}, \gamma_{\text{weak}}$ can be used to make up for undertrained auxiliary models. We achieve similar FID with other checkpoints and adjusted routing rates.

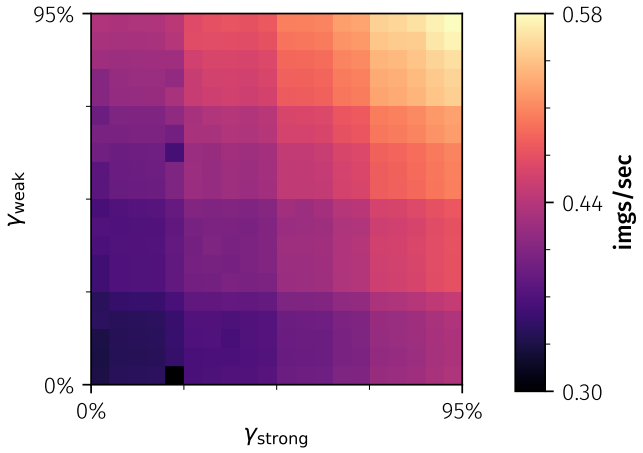

Figure A1. Inference speed for the guided setting. Lower left corner with zero γ_{strong}, γ_{weak} resembles naive guided inference. Introducing sparsity (Sparse Guidance) allows for drastically improved throughput showcased by brighter colors towards the top right corner.

B.2. Sparse Guidance in Large Scale T2I Models

In Table 4 we show that applying our proposed Sparse Guidance to scaled T2I models yields better performance than CFG. Additionally, Sparse Guidance enables faster inference as seen in Figure A1 where a grid over the γ_{strong},γ_{weak} with a 0.05 stepsize is shown.

GenEval [18] For GenEval (see Table 5), we stack our proposed Sparse Guidance method on top of Classifier-free Guidance and utilize $\omega = 2.5, \gamma_{\text{strong}} = 0.2$ and $\gamma_{\text{weak}} = 0.7$.

HPSv3 [47] For the HPSv3 score (see Table 4), we follow the proposed benchmark in Ma et al. [47] with identical prompts. We utilize Sparse Guidance with $\omega = 1.8, \gamma_{\text{strong}} = 0.1$ and $\gamma_{\text{weak}} = 0.8$.

C. Auxiliary MAE loss under Flow Matching

To facilitate a fair comparison between our SiT [43] baseline and MaskDiT [74], we derive the MaskedAutoEncoder (MAE) loss for the flow-matching objective (see Table 1, Figure 8). MaskDiT [74] combines a score-matching loss on visible tokens with a masked reconstruction (MAE) objective on masked tokens in diffusion models. We generalize this formulation to the *flow-matching* objective. Let $\mathcal{I}$ denote the token index set and $\mathbf{M} \in \{0,1\}^{\mathcal{I}}$ a random binary mask (1 for masked, 0 for visible). We define the visible mask as $\bar{\mathbf{M}} = \mathbf{1} - \mathbf{M}$. Following [74], the masked reconstruction loss is:

$$\mathcal{L}_{\text{MAE}} = \mathbb{E}_{x\sim p_{\text{data}}}\mathbb{E}_{t\sim[0,1]}\mathbb{E}_{\mathbf{M}}\big\|\big(D_\theta(x_t \odot \bar{\mathbf{M}}, t) - x\big)\odot \mathbf{M}\big\|^2, \quad \text{(A1)}$$

where D_θ predicts the denoised image at time t and $\odot$ denotes the Hadamard product. Unlike diffusion models, which predict the score $\nabla_{x_t} \log p_t(x_t)$, flow matching directly parameterizes the instantaneous displacement of particles along this trajectory. Given the path definition in Eq. 1, the latent states satisfy

$$x - x_t \;=\; (1-t)(x-z) \;=\; (1-t)\, v^\star(x_t, t), \quad \text{(A2)}$$

where $v^\star(x_t, t)$ is the oracle velocity field driving the transformation from z to x. This relation reveals that reconstructing a future state x_t from a clean sample x is equivalent to estimating the target velocity $v^\star(x_t, t)$ up to the scalar factor $(1-t)$. Hence, in the flow-matching formulation, masked reconstruction can be interpreted as learning to predict the intermediate flow direction that transports partially visible tokens toward their clean targets. Replacing $v^\star$ by its learned approximation v_θ, we have

$$D_\theta(x_t, t) - x_t \approx (1-t)\, v_\theta(x_t, t).$$

Consequently, the masked reconstruction term restricted to masked tokens can be reformulated as:

$$\begin{aligned}\mathcal{L}_{\text{MAE}} &= \mathbb{E}_x\mathbb{E}_{t\sim[0,1]}\mathbb{E}_{\mathbf{M}}\big\|(1-t)\, v_\theta(x_t \odot \bar{\mathbf{M}}, t)\odot \mathbf{M}\big\|^2 \\ &= \mathbb{E}_x\mathbb{E}_{t\sim[0,1]}\mathbb{E}_{\mathbf{M}}(1-t)^2\big\|v_\theta(x_t \odot \bar{\mathbf{M}}, t)\odot \mathbf{M}\big\|^2.\end{aligned} \quad \text{(A3)}$$

The overall training objective combines the standard flow-matching loss with the auxiliary masked reconstruction term. According to [32], routing models do not require additional auxiliary losses, so we use the standard flow matching objective. The final loss is defined as

$$\begin{aligned}\mathcal{L}_{\text{FM-mask}} = \mathbb{E}_{x,z,t}\Big[\big\|\,\bar{\mathbf{M}}\odot\big(v_\theta(x_t,t) - v^\star(x_t,t)\big)\,\big\|_2^2 \\ + \lambda\,\mathbb{E}_{x,t,\mathbf{M}}(1-t)^2\big\|v_\theta(x_t\odot\bar{\mathbf{M}},t)\odot\mathbf{M}\big\|_2^2\Big],\end{aligned} \quad \text{(A4)}$$

where λ balances the contribution of the masked reconstruction objective. In practice, we set λ empirically to ensure comparable magnitudes of the gradient between the two terms.

D. Qualitative Samples

We provide additional qualitative text-to-image results in Figure A2 and Figure A3, where we directly compare Classifier-Free Guidance (CFG) with Sparse Guidance (SG) in our TREAD-2.5B. Complementing these comparisons, Figure A4 presents a broader selection of SG-generated outputs. All text-to-image samples are produced using prompts sourced from the HPSv3 [47] benchmark subset.

Subsequently, Figure A5 and Figure A6 display ImageNet-256 results, contrasting unguided predictions, AutoGuidance (AG), CFG, and our SG method. Finally, Figure A7, Figure A8, and Figure A9 offer uncurated qualitative comparisons between SG_{FID} and SG_{FLOPS} to illustrate their respective visual characteristics.

Figure A2. Qualitative T2I examples comparing CFG to our proposed SG. Images with CFG tend to have more artifacts or seem blurry. SG provides crisp images with lower cost.

Figure A3. Qualitative T2I examples comparing CFG to our proposed SG. Images with CFG tend to have more artifacts or seem blurry. SG provides crisp images with lower cost.

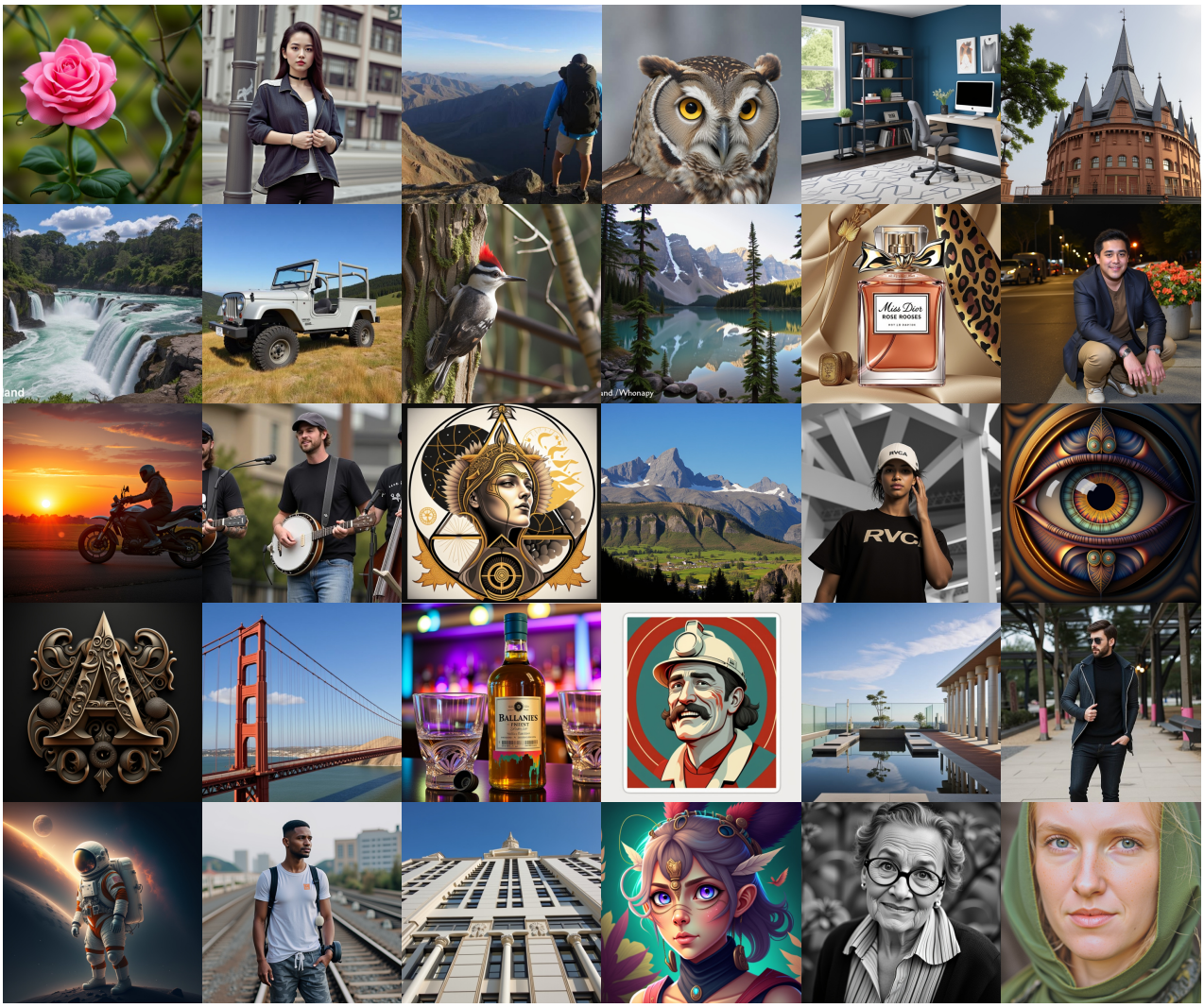

Figure A4. Additional T2I samples generated using Sparse Guidance. Prompts are taken from the HPSv3 benchmark subset.

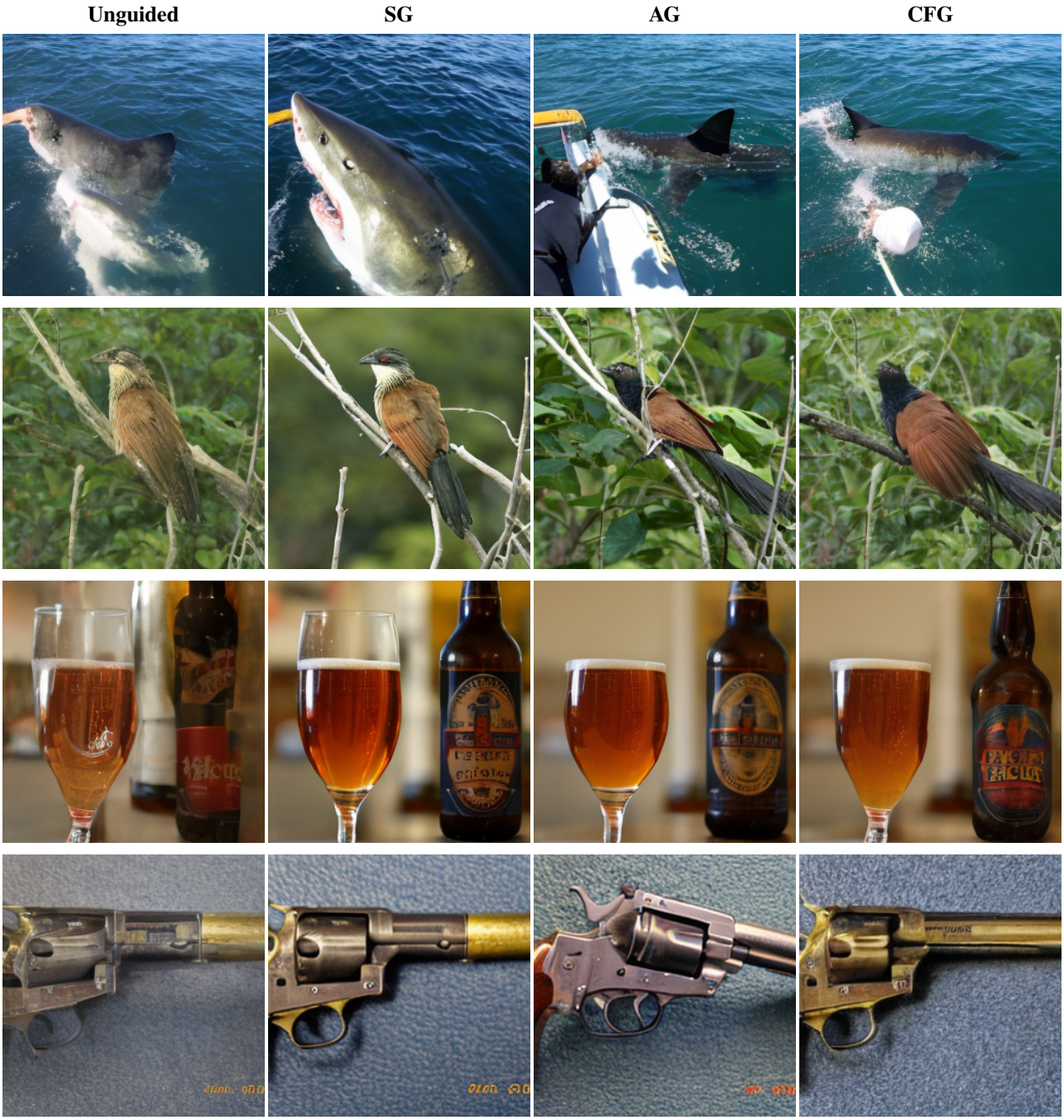

Figure A5. Qualitative samples from our ImageNet-256 model trained with token routing using a guidance scale of $\omega = 2.5$ across different methods: Unguided, Sparse Guidance (SG), AutoGuidance (AG), and Classifier-Free Guidance (CFG).

Figure A6. Qualitative samples from our ImageNet-256 model trained with token routing using a guidance scale of $\omega = 2.5$ across different methods: Unguided, Sparse Guidance (SG), AutoGuidance (AG), and Classifier-Free Guidance (CFG).

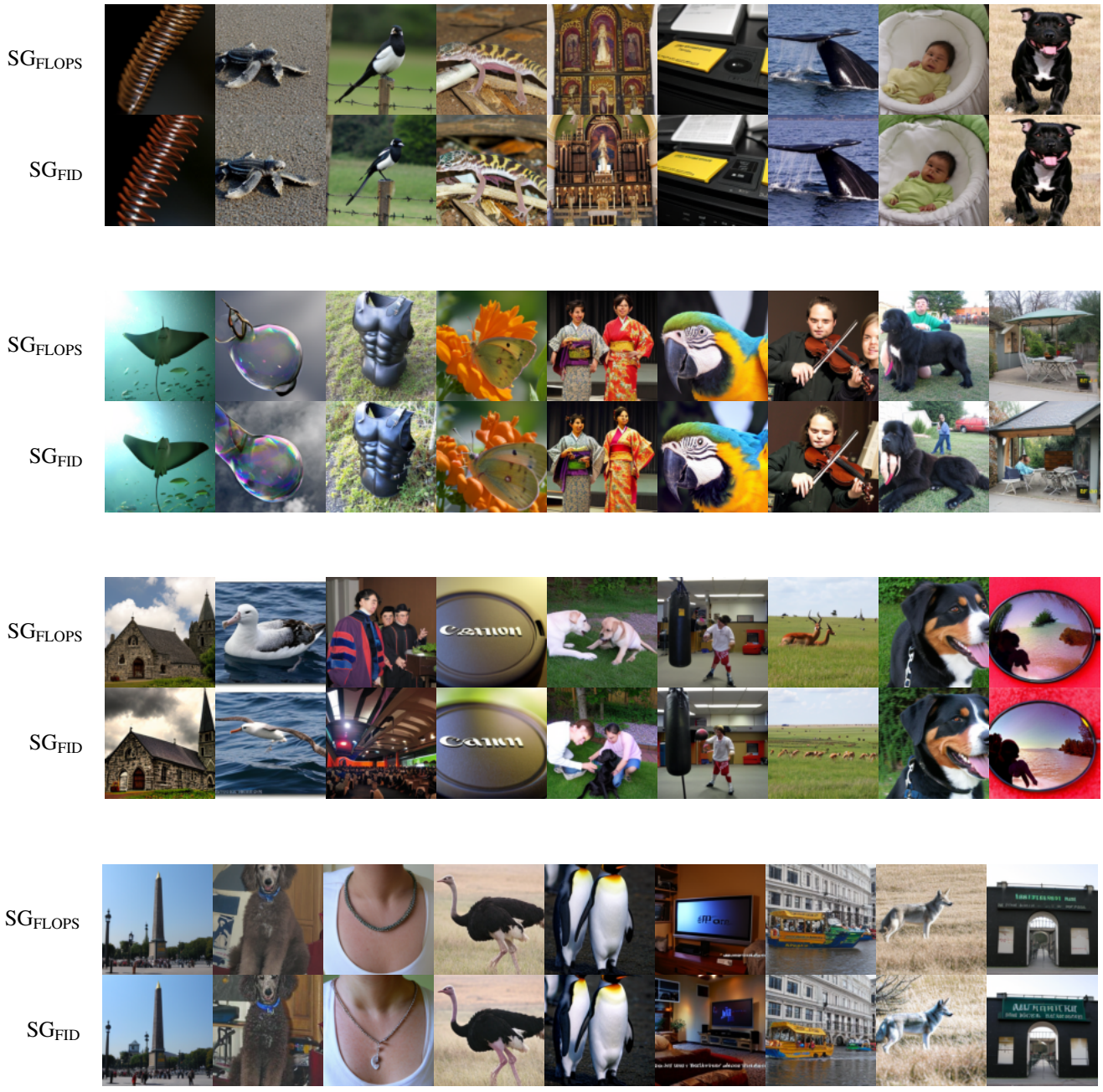

Figure A7. **Uncurated samples** of SG_{FLOPS} (top) and SG_{FID} (bottom) using $\omega = 2.5$ generated by our ImageNet-256 token routing model.

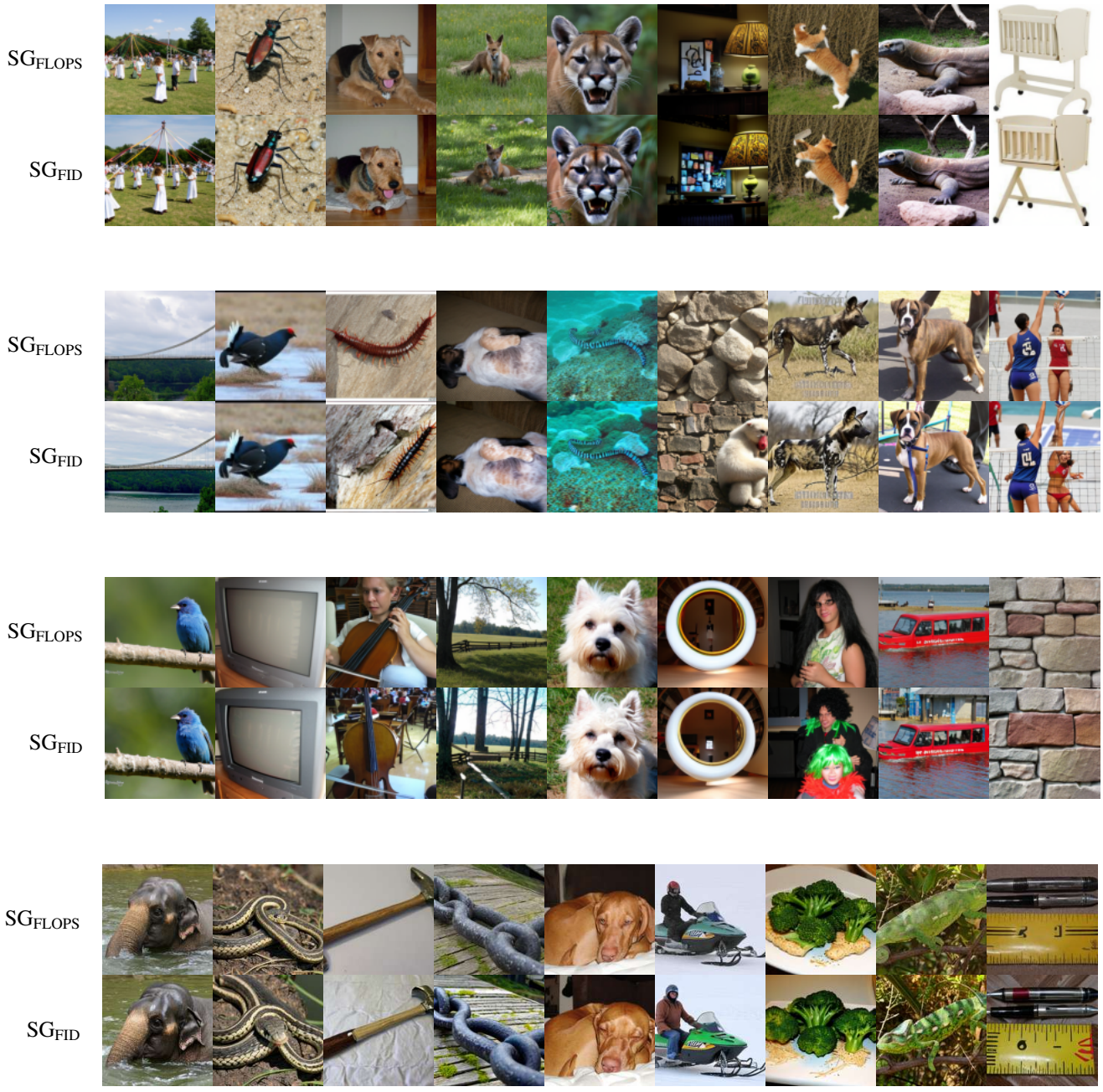

Figure A8. **Uncurated samples** of SG_{FLOPS} (top) and SG_{FID} (bottom) using $\omega = 2.5$ generated by our ImageNet-256 token routing model.

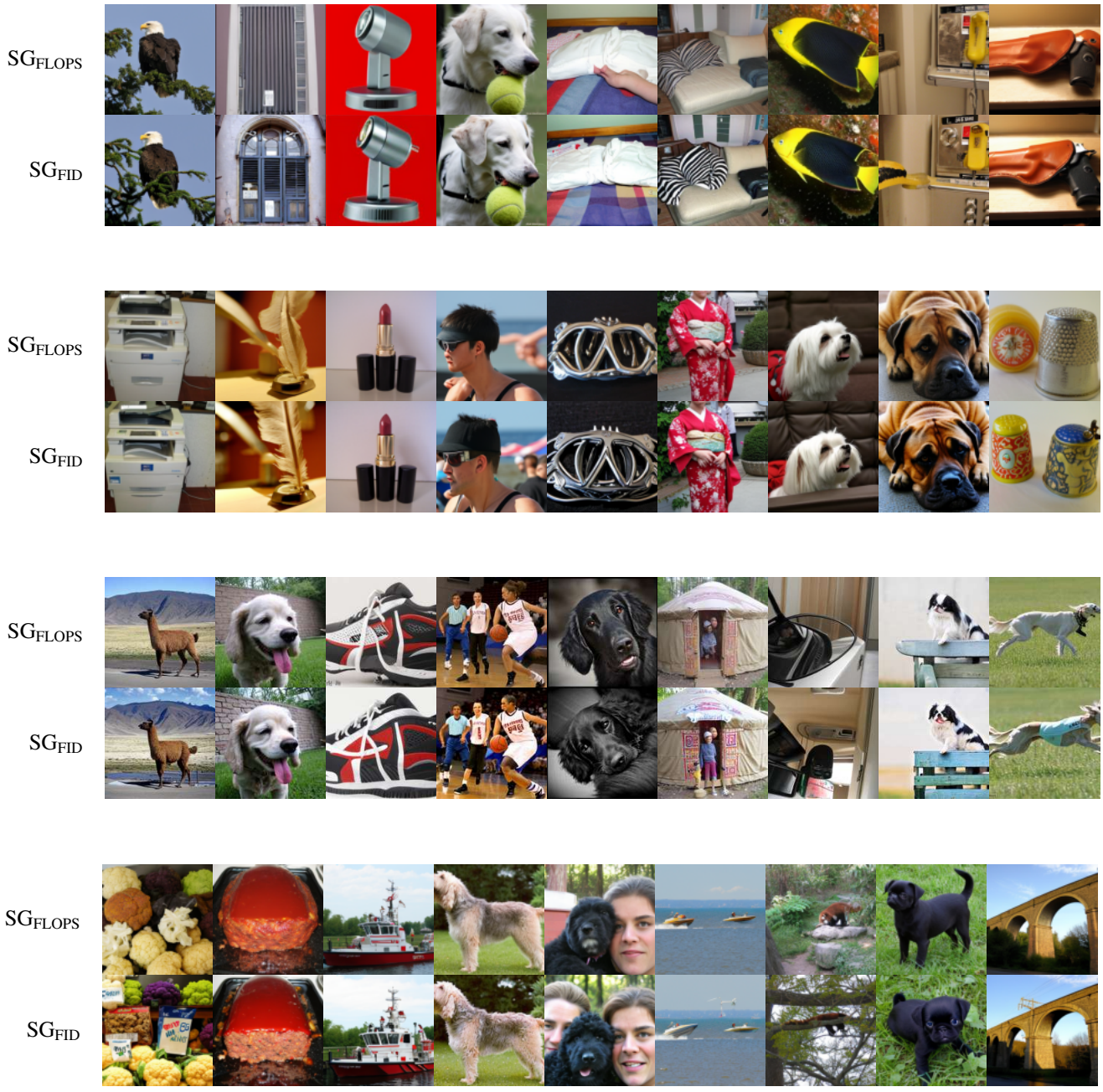

Figure A9. **Uncurated samples** of SG_{FLOPS} (top) and SG_{FID} (bottom) using $\omega = 2.5$ generated by our ImageNet-256 token routing model.