One Unified Representation: Resolving the Appearance-Semantics Dilemma via Structural Regularization

Ruiqi Yang[1,*] **Fei Zhou**[2,*,‡] **Yi Zhang**[3] **Liang Li**[3] **Xiangyu Yue**[2,4,✉]
[1]Zhejiang University [2]MMLab, The Chinese University of Hong Kong
[3]China Mobile Research Institute, Beijing, China [4]Shenzhen Loop Area Institute

Abstract

The Representation Autoencoder (RAE) shows that a frozen high-dimensional semantic ViT representation supports both image reconstruction and DiT-based generation. However, its frozen encoder omits appearance details needed for more accurate reconstruction. Unconstrained full fine-tuning can recover those details but catastrophically damages the semantic latent. In our ImageNet 256×256 evaluation, linear-probe accuracy falls from 83.8 to 10.6 and generation FID (gFID) rises from 2.40 to 20.95, even as rFID improves to 0.05. This trade-off motivates encoder adaptation that preserves the structure of the pretrained representation. We propose ***S**emantic **S**tructure **R**egularization* (**SSR**), a self-distillation framework with a frozen semantic teacher and a student encoder adapted through LoRA. In addition to final-layer latent alignment, we introduce two complementary structural constraints. First, spatial self-similarity distillation matches the patch-wise affinity patterns of student and teacher features. These patterns describe relationships among spatial tokens, providing a target for relational structure while allowing the student to encode additional appearance information. Second, structure-anchored alignment supplies an explicit structural target from a different view of the image. A Fourier-domain high-pass transform emphasizes edges and boundaries while suppressing low-frequency appearance, and the teacher's features of this view guide the student's features of the original image. Together, these constraints allow reconstruction-driven adaptation while limiting degradation of the teacher's semantic organization. On ImageNet 256×256, SSR improves reconstruction over a matched frozen RAE baseline (rFID 0.59 to 0.38, PSNR 19.27 to 22.26), matches its linear-probe accuracy (83.8), keeps dense-task performance close, and modestly improves unguided gFID (2.40 to 2.30). The result is one latent that supports reconstruction, perception, and generation at a chosen operating point, rather than maximizing reconstruction alone. Code is available at https://github.com/reid-yeung/SSR.

1 Introduction

Visual representations lie at the heart of three core objectives: high-fidelity reconstruction, discriminative perception, and efficient generative modeling. Separate paradigms have historically served these objectives. Appearance-focused autoencoders, including the VAE series [16, 22, 49] and VQ-based tokenizers [10, 50, 57], encode pixel-level details for reconstruction but carry limited semantic content, while semantics-focused encoders such as the DINO family [3, 30, 41] and SigLIP [48] learn rich structural and semantic representations that excel at perception yet discard the appearance details needed for reconstruction. This long-standing split raises a fundamental question: *Can a single representation space unify all three objectives without sacrificing any of them*?

[*]Equal contribution. [‡]Project leader. [✉]Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

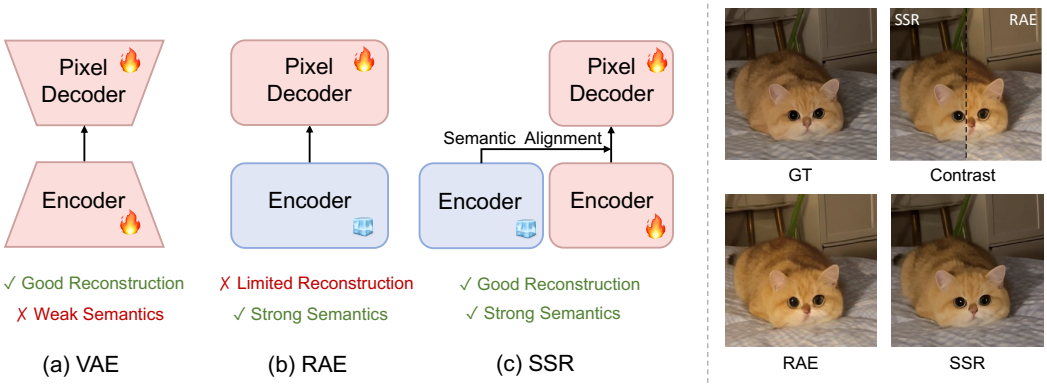


Figure 1: **Comparison of autoencoder paradigms.** (a) VAE jointly trains the encoder and pixel decoder for reconstruction, yielding good reconstruction but weak semantics. (b) RAE uses a frozen representation encoder, preserving semantics but limiting reconstruction fidelity. (c) SSR adapts the encoder with semantic alignment from a frozen teacher. The examples on the right show that SSR produces higher-quality reconstructions than RAE.

Latent diffusion models [1, 33, 36] leverage VAE latents as a compact perceptual space for high-fidelity image reconstruction, which significantly improves the efficiency of diffusion-based image generation. However, these reconstruction-oriented latents tend to preserve appearance details rather than high-level semantics (see Figure 1), and thus often require additional alignment losses to improve semantic controllability during generation [25, 52]. Semantic ViT representations [30, 41, 48], by contrast, encode rich structural and relational information that transfers broadly across perception tasks, but they discard appearance details by design, making direct image reconstruction from these features difficult. To reconcile these paradigms, the Representation Autoencoder (RAE) [60] bridges this gap by showing that a frozen semantic ViT representation can serve as the latent space for both a pixel decoder and a DiT-based generative model [32], achieving remarkable generation quality while preserving semantic discriminability. However, RAE's reconstruction quality remains fundamentally limited. Because the semantic encoder is strictly frozen, its latents inherently lack the fine-grained appearance details needed for high-fidelity pixel reconstruction.

The most direct remedy is to fully fine-tune the encoder jointly with a reconstruction loss, allowing it to capture appearance information. Without a semantic constraint, this causes catastrophic collapse in our ablation, with linear-probe accuracy falling from 83.8 to 10.6 and gFID rising from 2.40 to 20.95, even as rFID improves to 0.05 (Table 6). Adding only final-layer latent alignment avoids this global collapse but does not fully protect the adapted representation. Our Naive Fine-tune baseline has LP 82.5 and gFID 2.74, versus 83.8 and 2.40 for the frozen encoder, and its NYU depth RMSE rises from 0.373 to 0.425 (Table 1). These observations motivate preserving relational structure during appearance adaptation, beyond matching final-layer features alone.

In this paper, we propose ***S**emantic **S**tructure **R**egularization* (**SSR**), a training framework that improves reconstruction while keeping perception close to the frozen encoder and modestly improving generation in the same latent space. Our innovation lies in decoupling appearance learning from structural preservation by applying two complementary regularizers within a self-distillation framework (see Figure 2). Specifically, SSR adopts a frozen semantic teacher and a learnable student encoder, where the student is adapted via LoRA [17] to capture appearance details while remaining anchored to the pretrained semantic space.

Beyond basic latent distillation, we introduce *spatial self-similarity distillation*, which constrains the student to reproduce the teacher's patch-wise affinity matrix rather than its feature vectors. This patch-wise relational structure captures object layout, part correspondence, and scene organization, and is less tied to low-level appearance variations than direct feature matching. As a result, the student can preserve semantic geometry through the teacher's self-similarity structure while retaining the capacity to encode appearance details in complementary feature directions.

However, affinity constraints alone cannot prevent structural erosion in directions the affinity matrix does not capture. We therefore introduce *structure-anchored alignment*. A high-pass filter via the Fourier transform extracts a structural map that retains edges and contours while suppressing

Table 1: **Comparison of reconstruction, perception, and generation performance.** SSR improves reconstruction fidelity over RAE while preserving semantic discriminability across perception tasks. Naive Fine-tune denotes full encoder fine-tuning with reconstruction loss and basic latent distillation ($\mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{str0}}$) only, without structural regularization.

Model	Reconstruction			Perception				Generation			
	rFID↓	PSNR↑	SSIM↑	LP↑	ADE20K↑	NYU↓	NAVI↑	gFID↓	IS↑	Prec.↑	Rec.↑
RAE [60]	0.59	19.27	0.487	83.8	51.8	0.373	58.8	2.40	196.57	0.82	0.60
Naive Fine-tune	0.21	25.92	0.761	82.5	50.9	0.425	58.0	2.74	186.16	0.78	0.62
SSR	**0.38**	**22.26**	**0.607**	**83.8**	**51.4**	**0.389**	**58.7**	**2.30**	**201.36**	**0.81**	**0.60**

color and illumination, and the teacher's representation of this map serves as an explicit structural target for the student's representation of the original image. This asymmetric design reduces direct constraints on appearance encoding, while encouraging the structural component of the student's representation to remain consistent with the teacher's structural view. In this way, the model can maintain reconstruction fidelity while better preserving semantic structure (see Table 6).

We instantiate SSR with DINOv3-B [41] and re-implement RAE under the same encoder setting for a fair comparison. As visualized in Figure 1 and quantitatively reported in Table 1, SSR achieves an rFID of **0.38** on ImageNet-1K 256×256 [7], reducing rFID by **35.6%** compared to the RAE baseline, and PSNR rises from 19.27 to 22.26. SSR matches the frozen encoder on linear probing (83.8), while dense-task results remain close but are slightly lower on ADE20K and NAVI and worse on NYU depth (Table 1). When used as the latent space for DiT-based class-conditional generation, SSR reaches an unguided gFID of **2.30**, a modest improvement over the matched RAE baseline (2.40). We use "unified" to mean that one latent supports all three tasks at this operating point. Stronger reconstruction is possible, but the full fine-tune ablation shows that optimizing it without structural constraints can destroy semantic and generative utility.

In summary, our main contributions are listed as follows:

(1) We characterize the appearance–semantics trade-off, showing that unconstrained reconstruction adaptation collapses semantic and generative performance, while final-layer alignment alone leaves partial degradation.

(2) We propose SSR, a self-distillation framework with two structural regularizers that anchor patch-wise relational patterns and decouple structural supervision from appearance encoding.

(3) On ImageNet 256×256, SSR improves reconstruction, approximately preserves downstream perception, and modestly improves generation within one latent space.

2 Related Work

Visual Autoencoders for Reconstruction and Generation. Visual autoencoders establish the foundation for image reconstruction and latent-space generation. VAE [22] and VQ-VAE [49] introduce continuous and discrete latent representations, respectively, while VQ-GAN [10] further incorporates adversarial training for sharper reconstructions. Latent diffusion models [36] leverage these compact latents for scalable high-quality synthesis, and many efforts [12, 27, 32, 61] have demonstrated strong generation performance within this paradigm. However, appearance-focused latents lack semantic structure, prompting recent methods [25, 52, 58] to inject alignment with pretrained ViT representations during diffusion training. While effective, such approaches treat semantic alignment as an external objective, while the encoder latent remains appearance-centric. Thus, the two objectives remain unaligned within a single representation.

Semantic Visual Representations and Unified Latent Spaces. Large-scale visual pretraining has produced powerful semantic encoders that transfer broadly across perception tasks. The DINO family [3, 30, 41] learns rich patch-level structural representations via self-distillation, yielding strong visual understanding capabilities. Language-supervised encoders such as CLIP [35] and SigLIP [48] further align visual features with text for broad zero-shot transfer. While masked image modeling approaches like MAE [13] are more reconstruction-oriented, contrastive and self-distillation encoders often suppress low-level appearance cues in favor of semantic invariance. Consequently, despite their perceptual strength, high-fidelity reconstruction directly from these semantic features remains challenging. RAE [60] bridges this gap by using a frozen semantic ViT encoder to provide latents

for both pixel decoding and DiT-based generation. Yet this frozen encoder fundamentally limits reconstruction quality, and fully fine-tuning it with a reconstruction loss can lead to catastrophic semantic collapse, which is the core conflict that SSR is designed to address. SVG [39] instead augments a frozen DINOv3 encoder with a residual branch for appearance detail. Concurrent RAEv2 [42] aggregates frozen encoder layers and combines RAE with REPA-based guidance. SSR instead adapts the semantic encoder itself while constraining its structure. These mechanisms may be combined, but we have not evaluated their combination.

3 Method

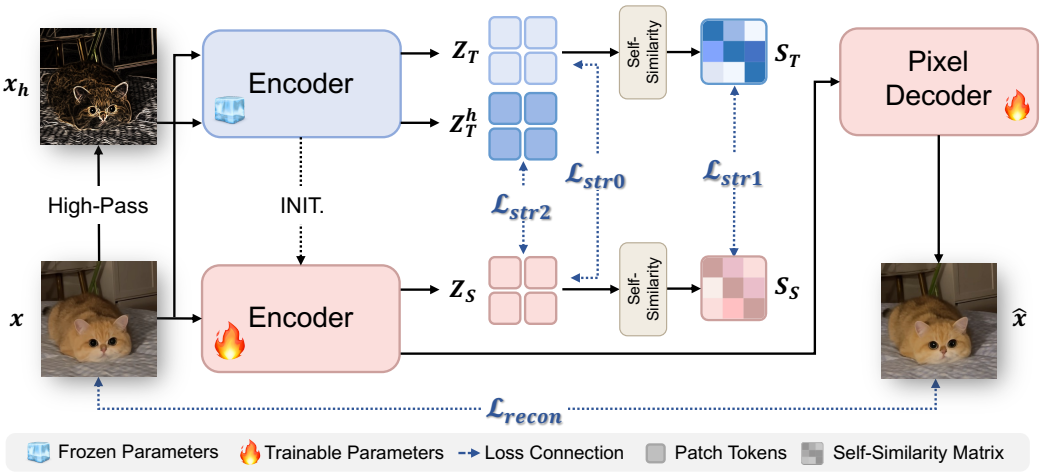


Figure 2: **Encoder adaptation under structure-preserving distillation.** A frozen teacher encoder provides two structural targets: the original-image latent and the high-pass structural latent. The student encoder, adapted via LoRA, is constrained by structure losses to preserve relational geometry.

3.1 Overview

We present ***S**emantic **S**tructure **R**egularization* (**SSR**), a framework that resolves the appearance–semantics conflict by decoupling appearance learning from structural preservation. SSR adopts a teacher-student self-distillation design with two complementary structural regularizers, enabling the encoder to capture appearance details without eroding the relational geometry of semantic features. SSR operates in two stages: a decoder warmup stage on a frozen encoder, followed by an encoder adaptation stage (see Figure 2) where a learnable student encoder is jointly trained with the decoder under structural regularization from a frozen teacher.

Given an input image $x \in \mathbb{R}^{H\times W\times 3}$, SSR builds on a representation autoencoder that maps x into patch-wise semantic latents $z = E(x) \in \mathbb{R}^{N\times d}$, where $N = (H/p)^2$ denotes the number of patch tokens and d the feature dimension. A pixel decoder D_ϕ reconstructs the image from the latent:

$$\hat{x} = D_\phi(E(x)) \tag{1}$$

Naively adapting E under reconstruction objectives improves appearance fidelity but distorts the relational geometry of the pretrained semantic space, motivating the structural constraints introduced below. Sections 3.2 and 3.3 detail the two structural regularizers, and Section 3.4 presents the full training objective.

3.2 Spatial Self-Similarity Distillation

A simple way to stabilize encoder adaptation is to align the student's final latent features with those of a frozen teacher encoder using an MSE loss. While this feature-level anchor helps prevent the student from drifting arbitrarily far from the pretrained semantic space, it does not explicitly constrain the relational structure among patch tokens. As a result, the student may still preserve individual feature responses while distorting the pairwise semantic geometry that encodes object layout, part

correspondence, and scene organization. We therefore introduce spatial self-similarity distillation, which regularizes the patch-wise affinity structure rather than only the absolute feature values.

Given student patch tokens $Z_S \in \mathbb{R}^{B\times N\times d}$ and teacher patch tokens $Z_T \in \mathbb{R}^{B\times N\times d}$ extracted from the same image x, we compute their respective patch-wise affinity matrices:

$$S_S = \bar{Z}_S \bar{Z}_S^\top \in \mathbb{R}^{B\times N\times N}, \quad S_T = \bar{Z}_T \bar{Z}_T^\top \in \mathbb{R}^{B\times N\times N} \tag{2}$$

where $\bar{Z} = Z/\|Z\|_2$ denotes ℓ_2-normalized patch tokens. The spatial self-similarity loss minimizes the MSE between the two affinity matrices:

$$\mathcal{L}_{\text{str1}} = \frac{1}{BN^2}\|S_S - S_T\|_F^2 \tag{3}$$

For hierarchical alignment, the loss is computed at multiple transformer layers $\mathcal{H} = \{l_1, l_2, \ldots\}$ and summed with progressively increasing layer-wise weights w_l that assign higher weight to deeper layers (see Appendix A.5 for details):

$$\mathcal{L}_{\text{str1}} = \sum_{l\in\mathcal{H}} \frac{w_l}{BN^2}\,\|S_S^{(l)} - S_T^{(l)}\|_F^2 \tag{4}$$

Unlike final-latent feature alignment, which provides only a coarse semantic anchor, relation matching explicitly constrains the pairwise geometry among patch tokens. This helps prevent reconstruction-driven adaptation from disrupting the teacher-induced semantic topology, leading to a more structurally faithful adapted representation.

3.3 Structure-Anchored Alignment

Spatial self-similarity distillation preserves relational structure but does not provide an explicit signal about which structural features the student should retain. Without additional guidance, the student may still allow structural information to erode in directions not captured by the affinity matrix. We address this by using the teacher's representation of a high-frequency structural view of the image as an explicit structural target. This view contains edges, contours, and shape cues while suppressing low-frequency color and illumination.

We construct a structural map x_h by applying a high-pass filter H to the input image in the frequency domain (see Appendix A.1 for details):

$$x_h = \left|\mathcal{F}^{-1}(H \cdot \mathcal{F}(x))\right| \tag{5}$$

The teacher processes x_h to produce structural patch tokens $Z_T^h = E_T(x_h)$, while the student processes the original image x to produce $Z_S = E_S(x)$. We align the two via patch-level cosine loss:

$$\mathcal{L}_{\text{str2}} = 1 - \frac{1}{BN}\sum_{b,n} \frac{Z_S^{(b,n)} \cdot Z_T^{h,(b,n)}}{\|Z_S^{(b,n)}\|\,\|Z_T^{h,(b,n)}\|} \tag{6}$$

By asking the student's representation of the original image to align with the teacher's representation of the structural map, we decouple structural supervision from appearance encoding. As with $\mathcal{L}_{\text{str1}}$, the loss is applied hierarchically across multiple layers.

The teacher's representation of the high-frequency view provides a structure-biased target that emphasizes local geometric patterns while reducing dependence on color and illumination. Aligning the student representation of the original image with this target encourages the adapted encoder to retain structure-sensitive semantics while avoiding the direct suppression of appearance information needed for reconstruction. This asymmetric design helps preserve the semantic structure inherited from the teacher while remaining compatible with reconstruction-driven adaptation.

3.4 Training Objective

Having introduced the two structural regularizers, we now describe how they are integrated into the training of the representation autoencoder. SSR trains the autoencoder in two stages, with the overall objective:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{SSR}} \tag{7}$$

In Stage 1, the encoder is frozen, and only the decoder D_ϕ is optimized under the reconstruction objective:

$$\mathcal{L}_{\text{recon}} = \mathcal{L}_{\text{L1}} + \lambda_{\text{LPIPS}}\,\mathcal{L}_{\text{LPIPS}} \tag{8}$$

where $\mathcal{L}_{\text{L1}}$ is the pixel reconstruction loss, as in RAE, and $\mathcal{L}_{\text{LPIPS}}$ [59] improves perceptual similarity in a deep feature space, with λ_{LPIPS} set to 1. This allows the decoder to adapt to the pretrained semantic latent space before any encoder modification, providing a stable initialization for Stage 2.

In Stage 2, we introduce a learnable student encoder E_S alongside the frozen teacher E_T, both initialized from DINOv3-B [41]. Rather than full fine-tuning, E_S is adapted via Low-Rank Adaptation (LoRA) [17], which constrains the adaptation to a low-dimensional subspace and limits drift from the pretrained semantic space. The student and decoder are jointly optimized under $\mathcal{L}_{\text{recon}}$ together with the semantic structure regularization term:

$$\mathcal{L}_{\text{SSR}} = \lambda_0\,\mathcal{L}_{\text{str0}} + \lambda_1(t)\,\mathcal{L}_{\text{str1}} + \lambda_2(t)\,\mathcal{L}_{\text{str2}} \tag{9}$$

where $\mathcal{L}_{\text{str0}}$ is a final-layer MSE distillation loss between student and teacher features with $\lambda_0 = 1$, serving as a global anchor that limits drift from the teacher's semantic space, and $\mathcal{L}_{\text{str1}}$ and $\mathcal{L}_{\text{str2}}$ are the spatial self-similarity and structure-anchored alignment losses described in Sections 3.2 and 3.3, respectively. The weights $\lambda_1(t)$ and $\lambda_2(t)$ follow a cosine warmup schedule to gradually introduce structural regularization during early adaptation. Their maximum values, 50 and 0.01, balance the differently scaled loss terms. The layer-wise weights emphasize deeper layers (Appendix A). The hierarchy, warmup, and high-pass cutoff choices are examined in Table 7, but we do not claim an exhaustive hyperparameter search.

In the later stage of Stage 2, we further add an adversarial loss $\mathcal{L}_{\text{adv}}$ to the reconstruction objective. Following RAE [60], we use a frozen DINO-S/8 [3] as the discriminator backbone and adopt the VQGAN adversarial objective [10] with adaptive loss weighting (see Appendix A.6 for details).

4 Experiments

4.1 Setup

Autoencoder training. SSR trains on ImageNet-1K [7] with images resized and center-cropped to 256×256. Training proceeds in two stages on 16 NVIDIA H100 GPUs with a batch size of 512 and bf16 precision. Stage 1 freezes the encoder and trains only the decoder for 12 epochs. Stage 2 continues from Stage 1 and jointly adapts the encoder via LoRA [17] and the decoder for 18 epochs. Training uses AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0) with a cosine learning rate schedule, a 1-epoch linear warmup, peak learning rate 2×10^{-4}, and minimum learning rate 2×10^{-5}. The LoRA adaptation targets the MLP up- and down-projection layers with rank $r = 32$, $\alpha = 64$, and dropout 0.05. All results are reported using EMA weights with decay 0.9978.

DiT training. For generation experiments, we adopt the DiT-DH-XL architecture and optimization setup from the RAE protocol [60]. The model uses a dual-head architecture with hidden sizes $[1152, 2048]$ and depths $[28, 2]$, operating on 16×16 latent patches of dimension 768. Unless otherwise noted, we train for 80 epochs with batch size 1024 using AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0), a linear warmup of 40 epochs, peak learning rate 2×10^{-4}, and cosine decay. The model uses flow matching with a linear path and velocity prediction, and we apply a logit-normal time distribution during training. At inference, we use 50-step Euler ODE sampling and report results using EMA weights with decay 0.9995.

We evaluate SSR on three tasks: reconstruction and generation on ImageNet-1K [7], and downstream perception across multiple benchmarks. Further training and evaluation details are provided in Appendices A and B, respectively. Compared with RAE, SSR adds 2.95M trainable parameters (415.35M to 418.30M, +0.71%). Under the same 30-epoch AE schedule, training cost rises from 170.5 to 194.7 GPU-hours (+14.20%), peak memory per GPU from 43.63 to 50.95 GiB (+16.78%), and measured step time from 0.511 to 0.632 s (+23.67%).

4.2 Reconstruction Quality

We compare SSR against discrete and continuous tokenizers on ImageNet 256×256 in Table 2, reporting PSNR and SSIM for pixel-level fidelity and rFID for perceptual distribution quality.



Figure 3: **Qualitative reconstruction comparison (no noise injection).** For each sample, we show the ground truth (top), RAE [60] reconstruction (middle), and SSR reconstruction (bottom). SSR produces sharper details and more faithful color rendition, with reduced color shift and better preservation of fine-grained textures. Red boxes highlight regions of interest for easier comparison.

Figure 3 shows qualitative comparisons between SSR and RAE. SSR achieves an rFID of 0.38, outperforming RAE by 0.21 and all discrete baselines, while also improving PSNR by 2.99 and SSIM by 0.12, indicating gains in both perceptual and pixel-level fidelity. These improvements are consistent with the qualitative results, in which SSR better preserves fine-grained structures, sharp boundaries, and color consistency than RAE. Without noise injection, SSR† further reaches an rFID of 0.29, with PSNR improved by 3.33 and SSIM by 0.14 over RAE†. This suggests that SSR improves reconstruction quality across both noisy and noise-free latent settings, demonstrating that structural regularization remains compatible with high-fidelity reconstruction.

Table 2: **Reconstruction performance comparison on ImageNet 256×256.** SSR improves over the matched RAE baseline across all reported metrics. † denotes no noise added to the latent. Red annotations indicate absolute improvements over the corresponding RAE baseline.

Model	Type	Ratio	rFID↓	PSNR↑	SSIM↑
LlamaGen [44]	Discrete	16	2.19	20.79	0.675
Show-o [56]	Discrete	16	3.50	21.34	0.590
Tokenflow [34]	Discrete	16	1.37	21.41	0.687
DualViTok [18]	Discrete	16	1.37	22.53	0.741
VILA-U [54]	Discrete	16	1.80	–	–
BLIP3o [4]	Continuous	16	3.28	14.71	0.58
Emu2 [45]	Continuous	14	3.27	13.49	0.423
Unilip [47]	Continuous	32	0.79	22.99	0.747
RAE [60]	Continuous	16	0.59	19.27	0.487
SSR	Continuous	16	**0.38**$^{-0.21}$	**22.26**$^{+2.99}$	**0.607**$^{+0.12}$
RAE† [60]	Continuous	16	0.44	20.51	0.543
SSR†	Continuous	16	**0.29**$^{-0.15}$	**23.84**$^{+3.33}$	**0.685**$^{+0.14}$

4.3 Downstream Perception

We evaluate the semantic quality of SSR's encoder across four tasks: ImageNet-1K [7] linear probing (top-1 accuracy) for global classification, ADE20K [62] mIoU for semantic segmentation, NYU Depth V2 [40] RMSE for monocular depth estimation, and NAVI [19] correspondence recall for 3D geometric understanding. As shown in Table 3, SSR performs comparably to DINOv3 [41] across all four tasks, while naive fine-tuning de-

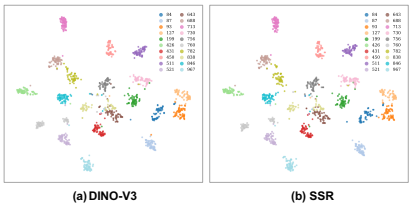


Figure 4: **t-SNE** of DINOv3 and SSR features.

grades performance on every metric. This confirms that SSR's structural regularization effectively preserves the relational geometry of the pre-trained encoder, even after joint adaptation with the decoder. The t-SNE visualization in Figure 4 further illustrates that SSR encoder features maintain well-separated semantic clusters comparable to DINOv3 [41], whereas naive fine-tuning weakens cluster separation.

Figure 5 further evaluates few-shot transfer across 10 diverse classification datasets following the CoOp [63] evaluation protocol, including StanfordCars [23], UCF101 [43], Caltech101 [11], Flowers102 [29], SUN397 [55], DTD [6], EuroSAT [14], FGVCAircraft [28], OxfordPets [31], and Food101 [2]. SSR remains competitive with DINOv3 across all datasets, demonstrating that structural regularization preserves transferable semantic representations well beyond the ImageNet training distribution.

Table 3: **Downstream perception performance comparison.** SSR preserves competitive semantic performance across classification and dense prediction tasks after adaptation for reconstruction. Red annotations denote improvement over Naive Fine-tune.

Model	Linear Probe ↑	ADE20K ↑	NYU ↓	NAVI ↑
SigLIP 2 [48]	–	41.6	0.512	25.4
DINOv2 [30]	83.0	48.4	0.416	56.9
DINOv3 [41]	83.8	51.8	0.373	58.8
Naive Fine-tune	82.5	50.9	0.425	58.0
SSR	**83.8**$^{+1.3}$	**51.4**$^{+0.5}$	**0.389**$^{-0.036}$	**58.7**$^{+0.7}$

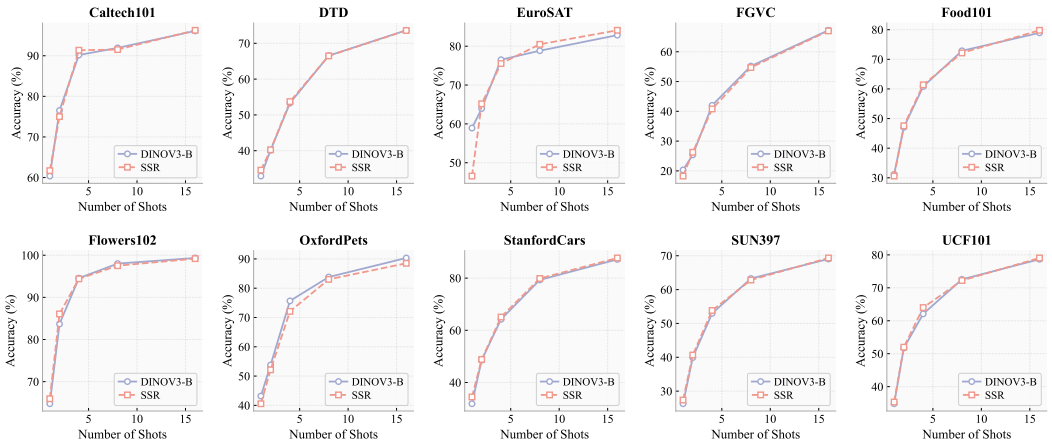


Figure 5: **Few-shot transfer accuracy across 10 classification datasets.** SSR is evaluated following the CoOp [63] protocol on 10 diverse datasets and compared against DINOv3 [41]. SSR remains competitive with DINOv3 across these datasets, demonstrating that structural regularization preserves transferable semantic representations well beyond the ImageNet [7] training distribution.

4.4 Generation from Learned Latents

We evaluate class-conditional generation on ImageNet 256×256 using a DiT-DH-XL model trained for 80 epochs on SSR latents, and report the results in Table 4. Without guidance, SSR achieves a gFID of 2.30, outperforming both RAE (2.40) and the naive fine-tune baseline (2.74). With AutoGuidance [20], SSR further reaches gFID 1.63, improving on RAE (1.75) and REPA-E (1.67) at the same 80-epoch training budget. The component ablation in Table 6 associates the generation gain with structural regularization rather than reconstruction fidelity alone. The improvement over matched RAE is modest relative to the additional AE cost reported in the setup.

The matched 280-epoch comparison in Table 4 retains SSR's guided gFID advantage (1.26 vs. 1.38), while its precision is slightly lower (0.79 vs. 0.80). This does not reproduce every setting of the original RAE's longer training regime. With DINOv2-B and SigLIP2-B teachers, SSR also improves both reconstruction and unguided generation relative to the matched RAE baselines (Table 5).

Table 4: **Generation performance comparison on ImageNet 256×256.** Our RAE/SSR rows use DINOv3-B. The 280-epoch rows report guided results only. Guided results use AutoGuidance [20] with scale 1.42. Dashes denote unreported metrics.

Model	#Params	Epochs	Generation@256 w/o guidance				Generation@256 w/ guidance			
			gFID↓	IS↑	Prec.↑	Rec.↑	gFID↓	IS↑	Prec.↑	Rec.↑
Pixel Diffusion										
PixNerd [51]	700M	160	–	–	–	–	2.15	297.0	0.79	0.59
PixelFlow [5]	677M	320	–	–	–	–	1.98	282.1	0.81	0.60
ADM [8]	554M	400	10.94	101.0	0.69	0.63	3.94	215.8	0.83	0.53
Latent Diffusion with VAE										
MaskDiT [61]	675M	1600	5.69	177.9	0.74	0.60	2.28	276.6	0.80	0.61
DiT [32]	675M	1400	9.62	121.5	0.67	0.67	2.27	278.2	0.83	0.57
SiT [27]	675M	1400	8.61	131.7	0.68	0.67	2.06	270.3	0.82	0.59
MDTv2 [12]	675M	1080	–	–	–	–	1.58	314.7	0.79	0.65
REPA-E [25]	675M	80	3.46	159.8	0.77	0.63	1.67	266.3	0.80	0.63
DDT [52]	675M	80	6.62	135.2	0.69	0.67	1.52	263.7	0.78	0.63
Latent Diffusion with RAE										
LV-RAE [26]	839M	800	2.42	223.8	0.77	0.64	1.82	249.7	0.75	0.67
RAE [60]	839M	80	2.40	196.57	0.82	0.60	1.75	219.44	0.81	0.62
Naive Fine-tune	839M	80	2.74	186.16	0.78	0.62	–	–	–	–
SSR	839M	80	**2.30**	**201.36**	**0.81**	**0.60**	**1.63**	**228.43**	**0.82**	**0.61**
RAE	839M	280	–	–	–	–	1.38	239.25	0.80	0.63
SSR	839M	280	–	–	–	–	**1.26**	**248.31**	**0.79**	**0.64**

Table 5: **Effect of teacher encoder on ImageNet** 256×256**.** RAE and SSR use matched settings within each teacher, and generation is evaluated after 80 epochs without guidance.

Teacher	Model	rFID↓	PSNR↑	SSIM↑	gFID↓	IS↑	Prec.↑	Rec.↑
DINOv2-B	RAE	0.57	19.01	0.480	2.45	216.9	0.82	0.58
DINOv2-B	**SSR**	**0.39**	**21.95**	**0.600**	**2.32**	**220.1**	**0.83**	0.58
SigLIP2-B	RAE	0.64	19.71	0.521	2.96	192.3	0.81	0.60
SigLIP2-B	**SSR**	**0.40**	**22.98**	**0.653**	**2.73**	**198.8**	**0.82**	0.60

SVG [39] uses a frozen DINOv3-S encoder with a residual branch and reports rFID 0.65, unguided gFID 3.36, and guided gFID 1.92 at 1400 epochs. RAEv2 [42] uses frozen DINOv3-L features, and its $K = 7$ configuration reports rFID 0.29, unguided gFID 1.65, and guided gFID 1.06 at 80 epochs. Our DINOv3-B SSR reports 0.38, 2.30, and 1.63, respectively, at 80 epochs. These absolute numbers are not controlled comparisons because encoder scale, training budget, sampling steps, and guidance methods differ. In particular, SSR does not surpass RAEv2's absolute guided gFID. Its contribution is regularized adaptation of the encoder itself. Combining these approaches remains untested.

4.5 Ablation Study

Encoder adaptation strategy. We compare encoder adaptation strategies in Table 6. Full Fine-tune and pure LoRA-MLP use $\mathcal{L}_{\text{recon}}$ alone, while Naive Fine-tune, LoRA-Attn, and the remaining LoRA rows also use $\mathcal{L}_{\text{str0}}$. Freezing the encoder preserves semantic quality but yields poor reconstruction. Unconstrained Full Fine-tune reaches rFID 0.05 while LP falls to 10.6 and gFID rises to 20.95. Naive Fine-tune retains a final-layer anchor and shows less severe, partial degradation, with LP 82.5 and gFID 2.74. Pure LoRA-MLP reaches rFID 0.28 and LP 83.0. With $\mathcal{L}_{\text{str0}}$, these become 0.31 and 83.3. The pure LoRA-MLP gFID was not measured. The anchored LoRA-MLP variant improves rFID and LP over LoRA-Attn and is used in the subsequent structural ablations.

Structural distillation. LoRA-MLP with $\mathcal{L}_{\text{str0}}$ matches frozen RAE's gFID of 2.40 while improving rFID to 0.31, so LoRA plus final-layer alignment alone brings no generation gain. Adding $\mathcal{L}_{\text{str1}}$ reduces gFID to 2.32 and raises LP from 83.3 to 83.6. Adding $\mathcal{L}_{\text{str2}}$ reduces gFID to 2.30 and raises LP to 83.8 (Table 6). Thus, the measured 2.40-to-2.30 generation gain comes from the two

Table 6: **Encoder adaptation and structural distillation.** "No distill." and Full Fine-tune use $\mathcal{L}_{\text{recon}}$ alone, while Naive Fine-tune, LoRA-Attn, and the subsequent LoRA rows include $\mathcal{L}_{\text{str0}}$.

Model	rFID ↓	LP ↑	gFID ↓
Frozen encoder	0.59	83.8	2.40
Full Fine-tune	0.05	10.6	20.95
Naive Fine-tune	0.21	82.5	2.74
LoRA-Attn	0.36	83.2	–
LoRA-MLP (no distill.)	0.28	83.0	–
LoRA-MLP (+$\mathcal{L}_{\text{str0}}$)	0.31	83.3	2.40
+$\mathcal{L}_{\text{str1}}$	0.38	83.6	2.32
+$\mathcal{L}_{\text{str1}}$+$\mathcal{L}_{\text{str2}}$ (SSR)	**0.38**	**83.8**	**2.30**

structural losses, despite rFID moving from 0.31 to 0.38. This is a deliberate operating point rather than a reconstruction ceiling. Naive Fine-tune reaches PSNR 25.92 but has worse LP and gFID than SSR (Table 1), while unconstrained Full Fine-tune reaches rFID 0.05 but loses most linear-probe accuracy (Table 6). Further adding $\mathcal{L}_{\text{str2}}$ (structure-anchored alignment) yields the full SSR with the best trade-off across all metrics, demonstrating that the two regularizers are complementary. $\mathcal{L}_{\text{str1}}$ preserves relational geometry globally, while $\mathcal{L}_{\text{str2}}$ provides an explicit structural target that prevents erosion in directions not captured by the affinity matrix. Regarding the loss functions, MSE gives the best balance of rFID and LP for $\mathcal{L}_{\text{str1}}$ (see Table 7) as it directly penalizes deviations in affinity matrix geometry, while cosine similarity works best for $\mathcal{L}_{\text{str2}}$ as it focuses on directional alignment toward the structural target rather than on magnitude matching.

Additional design choices. Hierarchical supervision across multiple transformer layers improves LP from 83.4 to 83.8 and gFID from 2.46 to 2.30 in Table 7, suggesting that constraints at different depths are complementary despite a modest rFID trade-off. Cosine warmup in the same table yields a better balance, improving LP from 83.6 to 83.8 and gFID from 2.34 to 2.30, as gradually introducing structural regularization lets the student first adapt to reconstruction before being constrained by the teacher-induced geometry. For the high-pass cutoff, Table 7 shows that 0.05 works best, indicating that an intermediate frequency band provides the most effective structural target. Too low a cutoff retains less relevant low-frequency appearance variations in the structural view, while too high a cutoff removes useful geometric cues.

Table 7: **Design-choice ablations.** Each variant changes one setting from SSR (MSE for $\mathcal{L}_{\text{str1}}$, cosine for $\mathcal{L}_{\text{str2}}$, hierarchy and warmup enabled, and cutoff 0.05).

Choice	Setting	rFID↓	LP↑	gFID↓
SSR (Ours)		**0.38**	**83.8**	**2.30**
$\mathcal{L}_{\text{str1}}$ loss	Cosine	0.41	83.6	2.37
	KL	0.37	83.7	2.34
$\mathcal{L}_{\text{str2}}$ loss	MSE	0.40	83.4	2.44
Hierarchy	×	0.32	83.4	2.46
Warmup	×	0.39	83.6	2.34
Cutoff	0.01	0.39	83.7	2.33
	0.1	0.38	83.6	2.34

5 Conclusion

We characterize the appearance–semantics trade-off in representation autoencoders. Unconstrained full fine-tuning collapses semantic and generative performance, while final-layer alignment alone leaves partial degradation. To address this, we propose Semantic Structure Regularization (SSR), which decouples appearance learning from structural preservation through self-distillation with two complementary structural regularizers. On ImageNet 256×256, SSR improves reconstruction over a matched frozen RAE baseline, matches its linear-probe accuracy, keeps dense-task performance close, and modestly improves generation quality at an additional AE training cost. These results show that one semantic latent can serve reconstruction, perception, and generation simultaneously at a chosen operating point. Optimizing reconstruction alone need not improve the other two objectives.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (No. 62306261), HK RGC-Early Career Scheme (No. 24211525) and ITSP Platform Project (No. ITS/600/24FP). This study was supported in part by the Centre for Perceptual and Interactive Intelligence, a CUHK-led InnoCentre under the InnoHK initiative of the Innovation and Technology Commission of the Hong Kong Special Administrative Region Government. This work is also partially supported by Hong Kong RGC Strategic Topics Grant (No. STG1/E-403/24-N and No. STG1/M-404/26-N), and CUHK-CUHK(SZ)-GDST Joint Collaboration Fund (No. YSP26-4760949).

References

[1] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014.

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[4] Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025.

[5] Shoufa Chen, Chongjian Ge, Shilong Zhang, Peize Sun, and Ping Luo. Pixelflow: Pixel-space generative models with flow. *arXiv preprint arXiv:2504.07963*, 2025.

[6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[9] Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21795–21806, 2024.

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[11] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[12] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Mdtv2: Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023.

[13] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[14] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

[15] 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. *Advances in neural information processing systems*, 30, 2017.

[16] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

[17] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

[18] Runhui Huang, Chunwei Wang, Junwei Yang, Guansong Lu, Yunlong Yuan, Jianhua Han, Lu Hou, Wei Zhang, Lanqing Hong, Hengshuang Zhao, et al. Illume+: Illuminating unified mllm with dual visual tokenization and diffusion refinement. *arXiv preprint arXiv:2504.01934*, 2025.

[19] Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, André Araujo, Ricardo Martin Brualla, Kaushal Patel, et al. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. *Advances in Neural Information Processing Systems*, 36:76061–76084, 2023.

[20] 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.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[23] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[24] 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.

[25] Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18262–18272, 2025.

[26] Siyu Liu, Chujie Qin, Hubery Yin, Qixin Yan, Zheng-Peng Duan, Chen Li, Jing Lyu, Chun-Le Guo, and Chongyi Li. Improving reconstruction of representation autoencoder. *arXiv preprint arXiv:2602.08620*, 2026.

[27] 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.

[28] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[29] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.

[30] 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.

[31] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[32] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[33] 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. *arXiv preprint arXiv:2307.01952*, 2023.

[34] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2545–2555, 2025.

[35] 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 *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[37] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[38] Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. In *International conference on machine learning*, pages 30105–30118. PMLR, 2023.

[39] Minglei Shi, Haolin Wang, Wenzhao Zheng, Ziyang Yuan, Xiaoshi Wu, Xintao Wang, Pengfei Wan, Jie Zhou, and Jiwen Lu. Latent diffusion model without variational autoencoder. *arXiv preprint arXiv:2510.15301*, 2025.

[40] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[41] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

[42] Jaskirat Singh, Boyang Zheng, Zongze Wu, Richard Zhang, Eli Shechtman, and Saining Xie. Improved baselines with representation autoencoders. *arXiv preprint arXiv:2605.18324*, 2026.

[43] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[44] Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

[45] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14398–14409, 2024.

[46] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[47] Hao Tang, Chenwei Xie, Xiaoyi Bao, Tingyu Weng, Pandeng Li, Yun Zheng, and Liwei Wang. Unilip: Adapting clip for unified multimodal understanding, generation and editing. *arXiv preprint arXiv:2507.23278*, 2025.

[48] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[49] Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/7a98af17e63a0ac09ce2e96d03992fbc-Paper.pdf.

[50] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[51] Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. *arXiv preprint arXiv:2507.23268*, 2025.

[52] Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer. *arXiv preprint arXiv:2504.05741*, 2025.

[53] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[54] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024.

[55] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[56] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

[57] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

[58] 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.

[59] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[60] Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

[61] Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *arXiv preprint arXiv:2306.09305*, 2023.

[62] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[63] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International journal of computer vision*, 130(9):2337–2348, 2022.

One Unified Representation: Resolving the Appearance-Semantics Dilemma via Structural Regularization

Supplementary Material

This supplementary material provides additional details to complement the main paper. Section A describes implementation details, including the Gaussian high-pass filter construction, model architecture, datasets, full loss configurations with training hyperparameters, and adversarial training setup. Section B elaborates on evaluation protocols for reconstruction, perception, and generation. Section C discusses current limitations and future directions, and Section D discusses broader impacts. Additional qualitative results, including 3D geometric correspondence (Figure 6), semantic segmentation (Figure 7), reconstruction, and generation samples, as well as a generation comparison of RAE, Naive Fine-tune, and SSR (Figure 10), are provided in Section E.

A Implementation Details

A.1 High-pass Filter

To explicitly extract high-frequency structure information from images, we apply a frequency-domain Gaussian high-pass filter as a preprocessing operation. Compared with the ideal high-pass filter and the Butterworth high-pass filter, the Gaussian high-pass filter yields a smoother transition in the frequency domain, thereby reducing ringing artifacts caused by abrupt frequency truncation. Given an input image batch $x \in \mathbb{R}^{B\times C\times H\times W}$, we first apply a two-dimensional fast Fourier transform (FFT) independently to each channel and shift the low-frequency components to the center of the spectrum. We then construct a Gaussian high-pass mask centered at the frequency origin:

$$H(u,v) = 1 - \exp\left(-\frac{D(u,v)^2}{2D_0^2}\right), \tag{10}$$

where $D(u,v)$ denotes the Euclidean distance from frequency location (u,v) to the spectrum center, and D_0 is the cutoff frequency. After filtering, we transform the frequency-domain result back to the spatial domain using the inverse FFT and take its absolute value as the final high-frequency response map. Before feeding it into the teacher model, we apply encoder normalization.

A.2 Architecture

We follow the overall architecture design of RAE [60], while replacing the original DINOv2-B [30] encoder with DINOv3-B [41]. Unlike DINOv2-B, DINOv3-B uses a patch size of 16×16 and produces latent tokens with channel dimension 768. We use an XL-size ViT decoder to reconstruct images from latent tokens, following the RAE decoder design. For semantic structure regularization, the student encoder is initialized from DINOv3-B, and a frozen DINOv3-B encoder is used as the teacher.

For image generation, we keep the diffusion model architecture consistent with previous work [32, 60], using DiT-DH-XL as the latent diffusion backbone and DiT-DH-S as the guiding model for AutoGuidance [20]. We use the checkpoint from the 14th epoch and set the AutoGuidance scale to 1.42. Unless otherwise specified, all generation and sampling settings follow RAE.

A.3 Datasets

We train the autoencoder and generative model on ImageNet-1K [7] at a resolution of 256×256. Following common practice in class-conditional image generation, we use the ImageNet-1K training set and report reconstruction and generation results on the validation set.

A.4 LoRA Configuration

We adapt the student encoder E_S via Low-Rank Adaptation (LoRA) [17] applied to the MLP up- and down-projection layers of all Transformer blocks in DINOv3-B, leaving self-attention weights and all bias terms frozen. We use rank $r = 32$, scaling factor $\alpha = 64$, and dropout 0.05. The low-rank

matrices follow standard initialization ($B = 0$, $A \sim \mathcal{N}(0, 1)$), so the LoRA branch contributes zero output at the start of Stage 2.

Table 8 justifies this choice. All variants are evaluated under $\mathcal{L}_{\text{str0}}$ only (i.e., with MSE distillation but without $\mathcal{L}_{\text{str1}}$ or $\mathcal{L}_{\text{str2}}$). Applying LoRA to MLP layers consistently outperforms attention-layer LoRA across all metrics, suggesting that MLP layers provide more appearance-relevant capacity than attention layers in this setting. Within MLP variants, $r = 32$ improves reconstruction quality over $r = 8$ (rFID 0.302 vs. 0.331, PSNR 23.70 vs. 23.48) while maintaining the same linear probe accuracy, indicating that the larger rank provides sufficient capacity for appearance adaptation without sacrificing semantic preservation. We therefore adopt LoRA-MLP with $r = 32$, $\alpha = 64$ as our final configuration.

Table 8: **Ablation of LoRA target layers and rank.** All models are trained with $\mathcal{L}_{\text{str0}}$ (MSE distillation) only. **Bold** denotes our chosen configuration.

Configuration	rFID ↓	PSNR ↑	SSIM ↑	LP ↑
LoRA-Attn ($r = 8$)	0.382	21.86	0.588	83.2
LoRA-Attn ($r = 32$)	0.354	22.43	0.615	83.2
LoRA-MLP ($r = 8$)	0.331	23.48	0.665	83.3
LoRA-MLP ($r = 32$)	**0.302**	**23.70**	**0.673**	**83.3**

A.5 Losses

The overall training objective consists of a reconstruction objective and the proposed semantic structure regularization:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{SSR}}. \tag{11}$$

We train the autoencoder in two stages. In the first stage, the encoder is frozen and only the decoder is optimized, so that the decoder can adapt to the latent space of the pretrained representation encoder. The objective in this stage consists of two reconstruction losses:

$$\mathcal{L}_{\text{recon}} = \mathcal{L}_{\text{L1}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}, \tag{12}$$

where the L1 loss enforces pixel fidelity, while the LPIPS loss [59] improves perceptual similarity in a deep feature space with a fixed weight $\lambda_{\text{LPIPS}} = 1$.

In the second stage, we fine-tune the encoder with LoRA [17] and jointly optimize the decoder. The training objective further includes our semantic structure regularization:

$$\mathcal{L}_{\text{SSR}} = \lambda_0\mathcal{L}_{\text{str0}} + \lambda_1(t)\mathcal{L}_{\text{str1}} + \lambda_2(t)\mathcal{L}_{\text{str2}}. \tag{13}$$

Here, $\mathcal{L}_{\text{str0}}$ is the basic latent distillation loss, implemented as an MSE loss between the last-layer features of the student and teacher encoders, with a fixed weight $\lambda_0 = 1$. $\mathcal{L}_{\text{str1}}$ denotes spatial self-similarity distillation, which constrains the patch-wise relational structure of the student encoder. We set its maximum weight to 50 and apply it hierarchically to multiple DINOv3 layers. $\mathcal{L}_{\text{str2}}$ denotes structure-anchored alignment, which further anchors the student representation to the teacher's structural prior. Its maximum weight is set to 0.01, and it is also applied hierarchically (see Table 9 for specific training configurations).

For $\mathcal{L}_{\text{str1}}$ and $\mathcal{L}_{\text{str2}}$, we apply a cosine warmup to their loss weights. This warmup prevents the structural constraints from dominating at the beginning of Stage 2, allowing the latent representation to progressively adapt to these regularizers. We additionally enable the GAN loss in the later stage of Stage 2 to improve reconstruction fidelity (see Appendix A.6 for details).

A.6 Adversarial Training

To enhance the perceptual quality of reconstructed images, we introduce an adversarial objective during the second stage of autoencoder training. We largely follow the adversarial training protocol of RAE [60], except for the epochs at which discriminator training and the adversarial loss begin. We employ a frozen DINO-S/8 [3] as the discriminator backbone, and both real and reconstructed images are resized to 224×224. For the discriminator optimization, we adopt the training losses

Table 9: **Training configuration of semantic structure regularization losses.**

Configuration	$\mathcal{L}_{\text{str0}}$	$\mathcal{L}_{\text{str1}}$	$\mathcal{L}_{\text{str2}}$
Loss weight	1	50	0.01
Warmup steps	–	5000	5000
Activated layers	Last layer	[6,8,10,12]	[8,10,12]
Layer-wise weights	–	[0.25,0.5,0.75,1.0]	[0.25,0.5,1.0]

used in StyleGAN-T [38], while for the autoencoder, we use the adversarial reconstruction loss following VQGAN [10]. The adversarial term is balanced with the reconstruction objective using the adaptive weighting strategy introduced in VQGAN. Optimization is performed with Adam [21], using cosine decay and a one-epoch warmup. We start updating the discriminator at epoch 18 and add the adversarial loss to the autoencoder objective at epoch 20.

B Evaluation Details

B.1 Reconstruction Evaluation

We evaluate reconstruction quality on the 50K images from the ImageNet-1K validation set using PSNR, SSIM [53], and FID [15]. PSNR measures pixel-level distortion, while SSIM evaluates structural similarity in terms of luminance, contrast, and local structure. For FID, we extract Inception-v3 features [46] from both reconstructed and real validation images and compute the FID between their feature distributions. This metric reflects the perceptual distribution gap of reconstructions. We report all reconstruction results using the autoencoder's EMA weights. More qualitative reconstruction results are provided in Figure 8.

B.2 Perception Evaluation

We evaluate the learned representations with linear probing on ImageNet-1K [7] and dense prediction on NYU Depth V2 [40], NAVI [19], and ADE20K [62]. We also evaluate few-shot classification across diverse datasets following CoOp [63].

Following the official DINOv3 implementation, we freeze the encoder and train a linear classifier on its global representations. We report top-1 classification accuracy on ImageNet-1K [7].

For relative monocular depth estimation on NYU Depth V2 [40], we perform dense linear probing. We train a linear head on the frozen patch output features, which are processed with layer normalization and a trained batch normalization layer. We report the Root Mean Squared Error (RMSE) metric to evaluate the structural and geometric quality of the representations.

For 3D geometric correspondence on the NAVI dataset [19], we adopt the Probe3D [9] evaluation protocol[1]. We use images resized to a side length of 512 pixels. Performance is measured by correspondence recall, which averages the recall across 3D distance error thresholds of 1 cm, 2 cm, and 5 cm.

For semantic segmentation on ADE20K [62], we evaluate the dense features using a linear probing protocol. Specifically, we freeze the visual backbone and train a linear classifier on the ADE20K training set. This linear layer is applied directly on top of the patch output features after layer normalization. To ensure stable training, these features are further normalized using a trained batch normalization layer before classification. We report the mean Intersection-over-Union (mIoU) as the primary evaluation metric.

For few-shot classification, we evaluate our model across 10 diverse image classification datasets: StanfordCars [23], UCF101 [43], Caltech101 [11], Flowers102 [29], SUN397 [55], DTD [6], EuroSAT [14], FGVCAircraft [28], OxfordPets [31], Food101 [2]. Following the standard evaluation protocol[2], models are trained using 1, 2, 4, 8, and 16 labeled examples per class. The models are subsequently evaluated on the full test set of each dataset. We report classification accuracy across these few-shot settings to demonstrate the generalizability and robustness of the learned representations

[1] https://github.com/mbanani/probe3d
[2] https://github.com/gaopengcuhk/Tip-Adapter

under limited downstream supervision. Detailed results for few-shot classification are summarized in Table 10.

Table 10: **Results of few-shot classification accuracy on 10 datasets.**

Model	Shots	Caltech101	DTD	EuroSAT	FGVC	Food101	Flowers	Pets	Cars	SUN397	UCF101
DINOv3-B	1	60.37	32.92	58.93	20.31	31.17	64.76	43.17	31.86	26.28	34.81
	2	76.51	40.13	63.94	25.38	47.05	83.68	53.77	48.56	39.92	51.73
	4	90.18	53.31	76.52	42.00	60.82	94.56	75.66	64.23	52.97	62.09
	8	91.93	66.55	78.84	55.18	72.87	98.01	83.81	79.32	63.26	72.56
	16	96.11	73.64	82.85	67.21	78.95	99.35	90.32	87.19	69.04	78.67
Naive Fine-tune	1	63.20	28.25	51.70	14.82	28.90	69.71	42.93	29.32	27.57	33.25
	2	81.78	42.67	59.67	24.84	45.36	83.72	47.07	42.35	40.02	50.62
	4	91.48	54.08	78.02	36.42	55.72	94.88	69.91	59.05	54.31	61.83
	8	91.16	66.73	73.68	49.14	69.49	97.40	79.31	74.67	62.86	72.01
	16	95.38	72.75	81.48	63.91	76.98	98.94	87.41	86.05	69.29	78.48
SSR	1	61.70	34.57	46.57	18.27	30.59	65.94	40.56	34.45	27.36	35.32
	2	75.01	40.25	65.17	26.25	47.55	86.07	52.14	48.82	40.66	51.97
	4	91.36	53.72	75.54	40.80	61.39	94.40	72.09	65.08	53.83	64.00
	8	91.52	66.49	80.49	54.70	72.16	97.52	83.02	79.87	62.82	72.22
	16	96.27	73.58	84.10	67.00	79.80	99.19	88.50	87.73	69.34	79.06

B.3 Generation Evaluation

We assess image generation quality on 50K generated samples using the ADM reference statistics [8]. The evaluation includes FID [15], Inception Score (IS) [37], precision, and recall [24]. FID compares the Inception-v3 feature distributions of generated and real images [46], while IS measures both the confidence and diversity of generated samples based on Inception-v3 predictions. Precision reflects the realism of generated images, and recall measures the coverage of the real data distribution. Additional qualitative generation results are shown in Figure 9.

C Limitations and Future Work

Our evidence for simultaneous reconstruction, perception, and generation is based on ImageNet 256×256, and several limitations remain. First, SSR relies on a frozen semantic teacher to provide structural supervision, so the semantic quality of the adapted representation is still largely tied to the teacher encoder. Exploring whether structural regularization can go beyond teacher preservation and further improve semantic representations is an important direction for future work. Second, the impact of structural regularization on the generative capacity of the latent space remains to be studied more comprehensively. Matched guided DiT experiments extend to 280 epochs, but do not cover every setting of the original RAE's longer training regime. We have not evaluated SSR at 512×512, so its high-resolution reconstruction and generation behavior remains unknown. The generation effect of pure LoRA-MLP without distillation has not been measured.

D Broader Impacts

This work aims to bridge semantic understanding and high-fidelity reconstruction within a unified visual representation. By reducing the appearance–semantics conflict in representation autoencoders, SSR may benefit image reconstruction, downstream perception, and DiT-based generation without requiring separate task-specific encoders. This could improve the efficiency, flexibility, and reusability of visual modeling pipelines, especially when both perception and generation are needed.

E Additional Visualizations



Figure 6: **Qualitative results on 3D geometric correspondence (NAVI).** SSR encoder features are used to establish dense correspondences across object instances under varying viewpoints and appearances.



Figure 7: **Qualitative results on semantic segmentation (ADE20K).** Each group shows the input scene (left), ground-truth segmentation mask (middle), and the prediction from SSR (right).

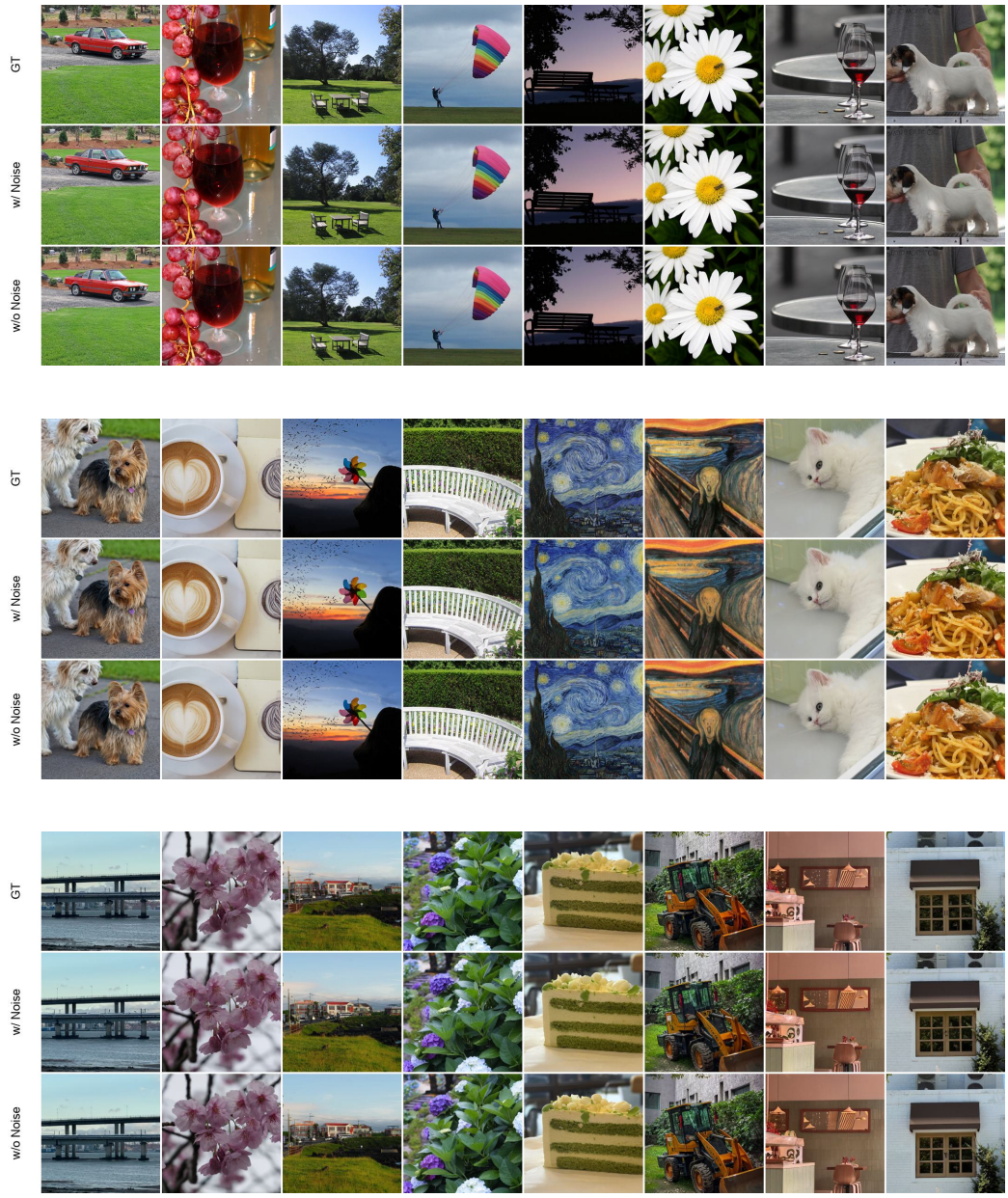


Figure 8: **Qualitative reconstruction results at 256×256 resolution.** *Row 1*: ground-truth images. *Row 2*: reconstructions with noise injection. *Row 3*: reconstructions without noise injection.

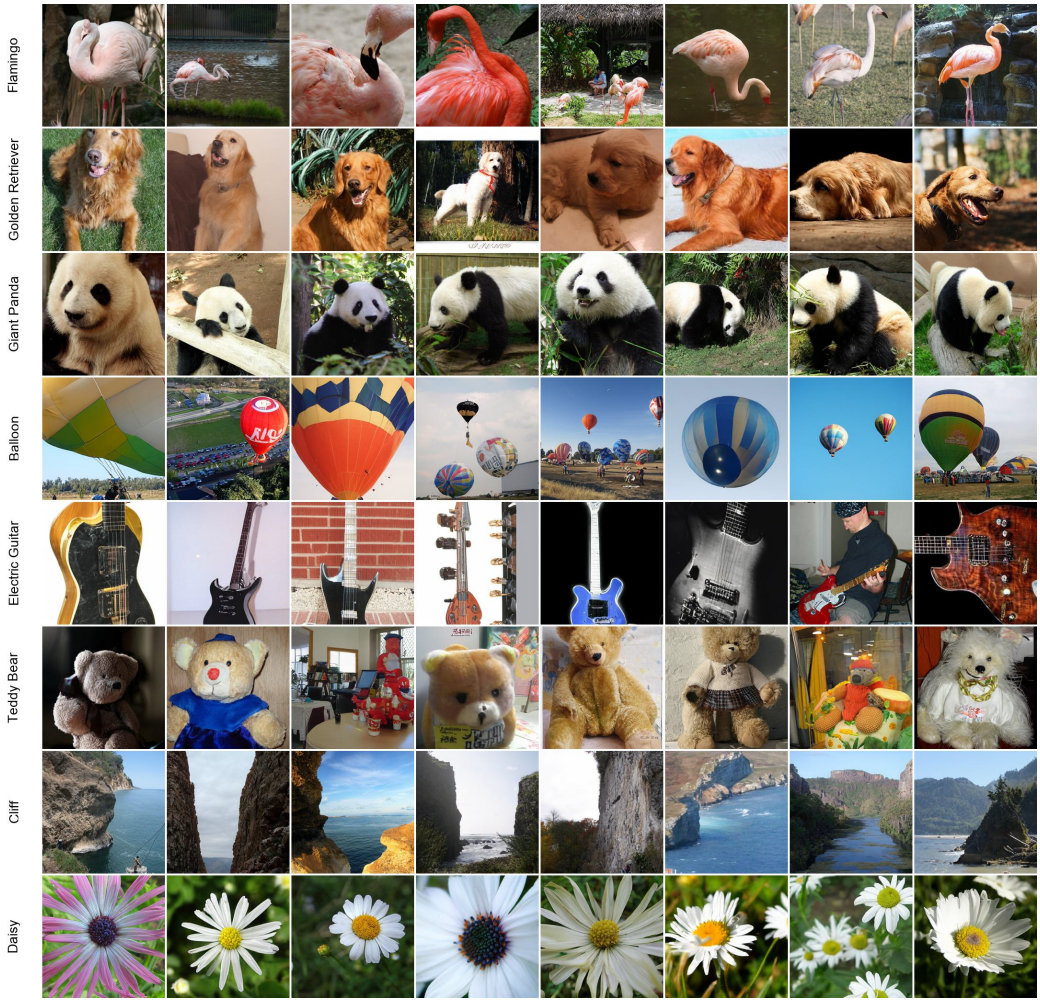


Figure 9: **Qualitative class-conditional generation results at 256×256 resolution.** Samples are drawn from diverse ImageNet categories using AutoGuidance with a DiT-DH-XL model trained on SSR latents.

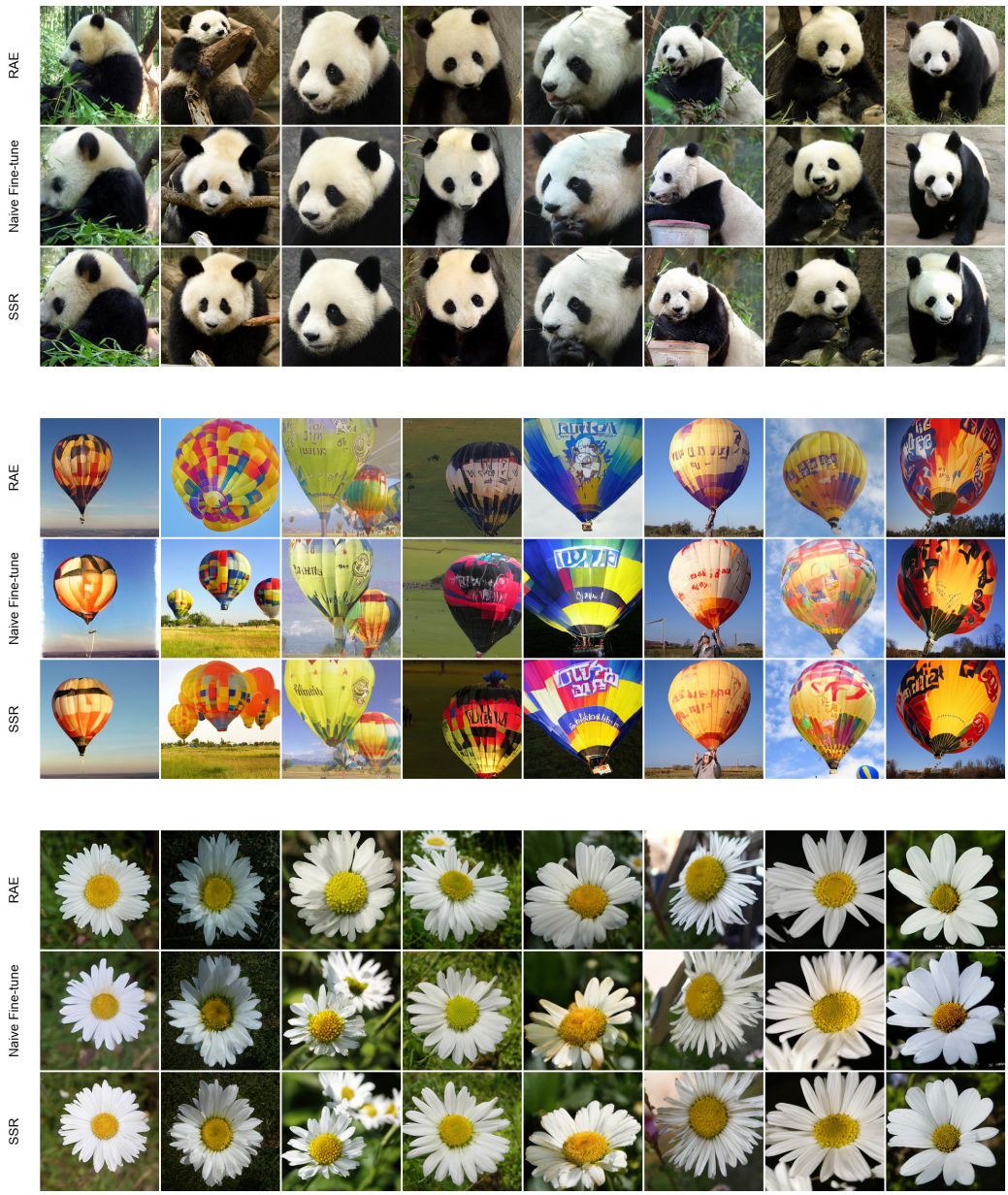


Figure 10: **Qualitative comparison of class-conditional generation at 256×256 resolution.** Each group shows samples drawn from DiT-DH-XL models trained for 80 epochs without guidance on RAE (top), Naive Fine-tune (middle), and SSR (bottom) latents.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main contributions and are consistent with the experimental results.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The limitations of the proposed method and its assumptions are discussed in the paper.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not include formal theoretical results requiring assumptions or proofs.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The paper provides detailed descriptions of the experimental setup, including hyperparameters, datasets, and evaluation metrics.

   Guidelines:

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

5. **Open access to data and code**

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

Answer: [Yes]

Justification: Implementation details are provided in the supplementary material, and the code repository is linked in the abstract.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: The paper provides detailed information about the experimental setup, including data splits, hyperparameters, and optimizer types.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [N/A]

Justification: The experiments focus on deterministic evaluations using standard benchmarks and metrics, so statistical significance analysis such as error bars is not applicable in this setting.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The computational resources and training setup are described in the paper.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research complies with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Potential positive and negative societal impacts are discussed.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The work focuses on representation learning and reconstruction on standard benchmarks and does not involve high-risk models or data that would require specific safeguards.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All datasets and external resources are properly cited with their licenses acknowledged.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

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

13. **New assets**

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

    Answer: [Yes]

    Justification: Newly introduced components are documented with sufficient details.

    Guidelines:

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

14. **Crowdsourcing and research with human subjects**

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

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or human subjects.

    Guidelines:

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

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

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

    Answer: [N/A]

    Justification: No human subjects were involved in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

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

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method does not use an LLM.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.