REPRESENTATION BY DESIGN IN GENERATION: CROSS-VIEW CLASS-TOKEN ALIGNMENT IN DIFFUSION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative and representation learning remain asymmetrically connected: semantic representations are used to improve diffusion generation, whereas the models' own representations are often treated as a by-product of synthesis. We ask whether diffusion models can instead be trained to learn substantially stronger semantic representations without sacrificing generation quality. SelfFlow takes a step in this direction by introducing self-supervised patch alignment into flow matching, but its main gains remain in faster convergence and improved generation. Inspired by DINO and iBOT, we extend this framework with cross-view class-token alignment to further strengthen semantic representations. Specifically, we form two independently noised, dual-timestep observations of each image and align each student class-token representation with the stop-gradient EMA-teacher target from the other observation. This objective is optimized jointly with the inherited flow-matching and local patch objectives. Notably, although the additional objective acts only on the class token, it strengthens both class-token and patch representations. Compared with a matched two-view baseline, ImageNet linear-probing accuracy improves by 9.4% using the class token and 10.1% using mean-pooled patch tokens, while frozen-backbone VOC2012 segmentation improves by 3.6 mIoU. These representation gains are achieved while maintaining comparable ImageNet generation FID. In text-to-image training, the same objective also improves generation FID, reducing it from 2.52 to 2.37 at matched checkpoints. Our results show that representation need not remain a by-product of generation or merely a tool for improving it: it can be directly optimized as a first-class capability of diffusion pretraining alongside generation.

1 INTRODUCTION

Generative and representation learning have become increasingly connected, but not symmetrically. Semantic representations are routinely introduced into diffusion models to accelerate training or improve generation, whether through feature alignment, semantic latent modeling, or representation-based autoencoders (Yu et al., 2025; Wu et al., 2025; Kouzelis et al., 2025; Zheng et al., 2026). In the other direction, pretrained diffusion models are commonly probed or adapted for correspondence, segmentation, and recognition (Baranchuk et al., 2022; Luo et al., 2023; Tang et al., 2023; Zhao et al., 2023; Yang & Wang, 2023), but their representations are often just treated as a by-product of generation. The asymmetry is therefore clear: representation learning is often introduced into generative training as a means to improve generation, whereas representation quality itself is rarely treated as an explicit end goal of generative pretraining. This raises a direct question: *can a diffusion model learn substantially stronger semantic representations without sacrificing generation quality?*

Recent methods begin to pursue both goals by incorporating self-supervision into diffusion training (Zhu et al., 2024; Jiang et al., 2026; Chefer et al., 2026). SelfFlow (Chefer et al., 2026), in particular, combines flow matching with alignment to cleaner EMA-teacher patch representations without relying on an external semantic encoder. This approach improves convergence and synthesis across modalities, but its reported peak linear-probing accuracy increases by only about one percentage point over the flow-matching baseline.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Inspired by DINO's cross-view class-token self-distillation and iBOT's usage of partial observations (Caron et al., 2021; Zhou et al., 2021), we extend this framework with cross-view class-token alignment to further strengthen semantic representations. For each image, we use two independent Gaussian noise realizations to form two dual-timestep observations. Each observation contains cleaner tokens that retain relatively complete image content and noisier tokens that provide partial information. To encourage a global representation of the image content shared across these observations, we prepend a CLS token and align each student class-token representation with the stop-gradient EMA-teacher target from the other noise branch. This alignment objective is optimized jointly with the inherited flow-matching and same-view patch objectives, which remain unchanged.

To ensure a fair comparison, the baseline and our method differ only in whether cross-view class-token alignment is active, as illustrated in Figure 1. Both use the same transformer architecture, added CLS token, independently noised dual-timestep inputs, EMA teacher, flow-matching objective, and local patch objective. Adding this single objective substantially improves ImageNet linear probing for both CLS and patch tokens. Patch-token accuracy increases by 10.1%, from 60.0% to 70.1%, while frozen-backbone semantic segmentation on VOC2012 improves by 3.6 mIoU, from 57.2 to 60.8 (Table 2). Both the patch-token probe and the segmentation head operate directly on frozen patch features without using the CLS readout. Thus, although the additional objective acts only on CLS, these gains reflect stronger patch representations rather than merely a better global readout. At the same time, ImageNet generation FID remains comparable to the baseline. In text-to-image training, the same objective also improves generation FID, reducing it from 2.52 to 2.37 at matched checkpoints.

Our results suggest that representation need not remain a by-product of generation or merely a tool for improving it. Instead, it can be optimized directly alongside synthesis as a first-class objective of diffusion pretraining, motivating future work on models that develop semantic and generative capabilities together.

2 Related Work

Semantic representations for diffusion generation. A growing line of work introduces semantic representations into diffusion models to improve optimization and synthesis. REPA aligns intermediate denoiser features with representations from a frozen self-supervised visual encoder (Yu et al., 2025). REG and ReDi incorporate semantics more directly into the generative state: REG jointly denoises image latents and a global representation token, while ReDi jointly models image latents and high-level visual features (Wu et al., 2025; Kouzelis et al., 2025). RAE goes further by replacing conventional VAE latents with the features of a pretrained representation encoder (Zheng et al., 2026). A parallel line seeks similar benefits without relying on an external encoder. SD-DiT introduces self-supervised discrimination into diffusion training, SRA aligns the model's own representations across layers and noise levels, and Self-Flow combines internal feature distillation with Dual-Timestep Scheduling for flow matching (Zhu et al., 2024; Jiang et al., 2026; Chefer et al., 2026). Conditioning Residuals instead feed compact summaries of internal features back through the denoiser's conditioning pathway (Xiang et al., 2026). Although these approaches differ in where and how semantics are introduced, representation learning is primarily used to accelerate generative training or improve synthesis.

Diffusion models as representation learners. A complementary line studies the semantic structure that emerges from diffusion models trained for generation. Intermediate denoising features have been extracted for label-efficient semantic segmentation, semantic correspondence, and dense visual perception (Baranchuk et al., 2022; Tang et al., 2023; Luo et al., 2023; Zhao et al., 2023). DDAE systematically evaluates the linearly separable features learned through unconditional diffusion pretraining, while DiffMAE modifies diffusion training with masked observations to prioritize downstream recognition (Xiang et al., 2023; Wei et al., 2023). More closely related to joint learning, D^3CL treats different diffusion noise levels as contrastive views when adapting a pretrained Stable Diffusion model (Dai et al., 2026). Together, these studies establish that diffusion models can support visual understanding, but the two directions remain uneven: semantic representations are commonly optimized when they benefit generation, whereas representations from generative models are more often probed, adapted, or transferred after generative pretraining. Our work fo-

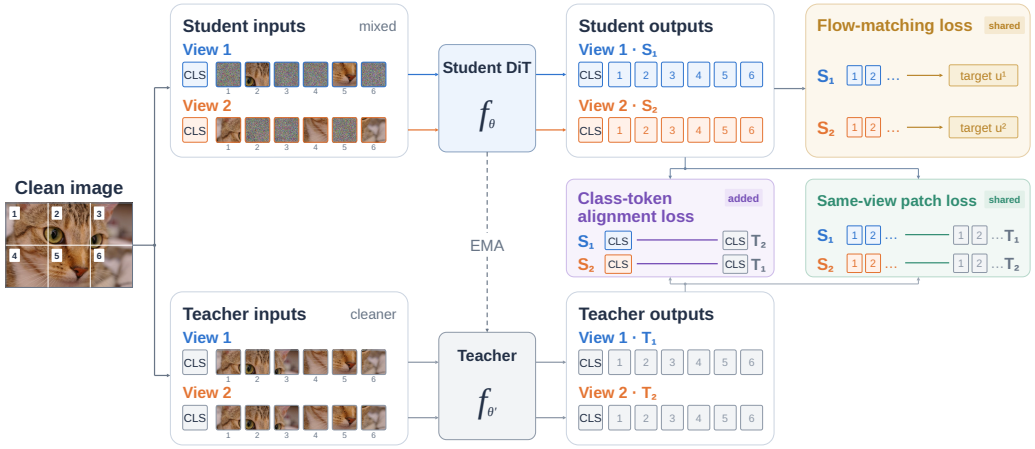


Figure 1: **Class-token alignment across diffusion views.** For each clean latent $\mathbf{z}$, two independent noise realizations $\boldsymbol{\epsilon}^{(1)}$ and $\boldsymbol{\epsilon}^{(2)}$ produce mixed-timestep student inputs and corresponding uniformly cleaner teacher inputs. The student f_θ and its exponential-moving-average (EMA) teacher $f_{\theta'}$ each process both views; S_v and T_v label their outputs for view v. Both variants share flow matching to the targets $\mathbf{u}^{(v)} = \mathbf{z} - \boldsymbol{\epsilon}^{(v)}$ and same-view patch alignment between (S_1, T_1) and (S_2, T_2). Our method additionally aligns class tokens across (S_1, T_2) and (S_2, T_1); the matched baseline sets $\lambda_{\text{cls}} = 0$. Numbered tokens indicate corresponding spatial patches. Teacher feature targets are stop-gradient, and the dashed arrow denotes the EMA update. The output boxes are schematic: alignment features may come from different blocks, and student projection heads are omitted.

cuses on the less explored setting in which semantic quality is itself a primary objective of diffusion pretraining and is evaluated alongside generation under a matched generative setup.

3 Method

Our goal is to introduce an explicit global semantic objective into SelfFlow (Chefer et al., 2026) while preserving its original generative and local representation objectives. Figure 1 summarizes the training pipeline. We first instantiate the dual-timestep process twice using independent noise, giving two observations of the same image. Both the baseline and our method use these observations, the same learnable class (CLS) token and EMA teacher, and the same flow and patch-level objectives. Our method adds only a cross-view alignment objective on the class token.

3.1 Two-View SelfFlow Baseline

Let $\mathbf{z} \in \mathbb{R}^{N \times d}$ denote a clean image latent represented by N patch tokens, where d is the latent dimension per patch. For a noise tensor $\boldsymbol{\epsilon} \in \mathbb{R}^{N \times d}$ with independent standard Gaussian entries, we use the linear transport path

$$q_t(\mathbf{z}, \boldsymbol{\epsilon}) = t\mathbf{z} + (1-t)\boldsymbol{\epsilon}, \qquad t \in [0, 1], \tag{1}$$

where larger t corresponds to a cleaner observation. The velocity along this path is constant:

$$\frac{\partial}{\partial t} q_t(\mathbf{z}, \boldsymbol{\epsilon}) = \mathbf{z} - \boldsymbol{\epsilon}.$$

For each training example, we sample $t_a, t_b \sim \mathcal{U}(0, 1)$ and define

$$t_{\text{c}} = \max(t_a, t_b), \qquad t_{\text{n}} = \min(t_a, t_b), \tag{2}$$

where t_{c} and t_{n} denote the cleaner and noisier timesteps, respectively. The same timestep pair is used for both views, while their Gaussian noise realizations and token masks are sampled independently.

For each view $v \in \{1, 2\}$, we sample $\boldsymbol{\epsilon}^{(v)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and independently draw $m_i^{(v)} \sim \text{Bernoulli}(\rho)$ for each patch index $i = 1, \ldots, N$. Here, ρ is the probability of selecting the cleaner state ($m_i^{(v)} =$

1). The mixed student input $\widetilde{\mathbf{z}}^{(v)}$ and its per-patch timesteps $\tau_i^{(v)}$ are given by

$$\widetilde{\mathbf{z}}_i^{(v)} = m_i^{(v)} q_{t_\mathrm{c}}\left(\mathbf{z}_i, \boldsymbol{\epsilon}_i^{(v)}\right) + \left(1 - m_i^{(v)}\right) q_{t_\mathrm{n}}\left(\mathbf{z}_i, \boldsymbol{\epsilon}_i^{(v)}\right), \tag{3}$$

$$\tau_i^{(v)} = m_i^{(v)} t_\mathrm{c} + \left(1 - m_i^{(v)}\right) t_\mathrm{n}. \tag{4}$$

We collect the per-patch timesteps into $\boldsymbol{\tau}^{(v)} = (\tau_1^{(v)}, \ldots, \tau_N^{(v)})$.

The student therefore receives a tokenwise mixture of relatively complete and partial information. The EMA teacher receives the uniformly cleaner observation from the corresponding noise branch:

$$\overline{\mathbf{z}}^{(v)} = q_{t_\mathrm{c}}\left(\mathbf{z}, \boldsymbol{\epsilon}^{(v)}\right). \tag{5}$$

Let f_θ denote the student diffusion transformer (DiT) and $f_{\theta'}$ its EMA teacher, with parameters θ and θ', respectively. Each network is shared across the two views. In Figure 1, S_v and T_v identify the student and teacher branches for view v. We write f_θ^{flow} for the student with its velocity-prediction head, giving

$$\widehat{\mathbf{U}}^{(v)} = f_\theta^{\mathrm{flow}}\left(\widetilde{\mathbf{z}}^{(v)}, \boldsymbol{\tau}^{(v)}, \mathbf{y}\right) \in \mathbb{R}^{N \times d} \tag{6}$$

where $\widehat{\mathbf{U}}^{(v)}$ is the patch-wise velocity prediction for view v and $\mathbf{y}$ is the class or text condition. The CLS-token output is not included in $\widehat{\mathbf{U}}^{(v)}$. We denote the corresponding velocity target by $\mathbf{u}^{(v)} = \mathbf{z} - \boldsymbol{\epsilon}^{(v)}$, shown as u^1 and u^2 in Figure 1. We optimize

$$\mathcal{L}_{\mathrm{flow}} = \frac{1}{2}\sum_{v=1}^{2} \mathrm{MSE}\left(\widehat{\mathbf{U}}^{(v)}, \mathbf{u}^{(v)}\right), \tag{7}$$

where MSE averages over patch and feature dimensions. Although the velocity target is constant along this linear path, the per-token timestep vector $\boldsymbol{\tau}^{(v)}$ remains an input to the transformer because it specifies the noise state of each patch.

As in SelfFlow, the baseline additionally aligns student patch features with cleaner EMA-teacher patch features. Let $\mathbf{P}_{\theta,\ell_s^p}^{(v)}$ denote student patch-token hidden features from block ℓ_s^p and $\mathbf{P}_{\theta',\ell_t^p}^{(v)}$ denote EMA-teacher patch-token hidden features from block ℓ_t^p. The subscript i indexes corresponding spatial patches. The local objective is:

$$\mathcal{L}_{\mathrm{patch}} = -\frac{1}{2N}\sum_{v=1}^{2}\sum_{i=1}^{N} \cos\left(g_{\mathrm{patch}}\left(\mathbf{P}_{\theta,\ell_s^p,i}^{(v)}\right), \mathrm{sg}\left[\mathbf{P}_{\theta',\ell_t^p,i}^{(v)}\right]\right), \tag{8}$$

where cos denotes cosine similarity, g_{patch} is a two-layer student projection head, and sg denotes stop-gradient. Patch alignment is *same-view*: the student and teacher features use the same noise realization and corresponding spatial positions. These flow and patch objectives define our two-view SelfFlow baseline.

3.2 Cross-View Class-Token Alignment

To introduce an explicit global semantic target, we prepend a learnable CLS token to the image-token sequence. Unlike patch tokens, the CLS token itself is not corrupted by the diffusion process. Its per-token timestep shift is set to zero, while it participates in the same transformer self-attention and global class or text conditioning as the image tokens. The same token architecture is used in both the baseline and our method.

Let $\mathbf{c}_{\theta,\ell^c}^{(v)}$ and $\mathbf{c}_{\theta',\ell^c}^{(v)}$ denote the student and EMA-teacher class-token representations from block ℓ^c. Inspired by class-token self-distillation in DINO and iBOT (Caron et al., 2021; Zhou et al., 2021), we define a cross-view alignment objective by matching each student representation to the EMA-teacher representation from the opposite view:

$$\begin{aligned}\mathcal{L}_{\mathrm{cls}} = -\frac{1}{2}\Big(&\cos\left(g_{\mathrm{cls}}\left(\mathbf{c}_{\theta,\ell^c}^{(1)}\right), \mathrm{sg}\left[\mathbf{c}_{\theta',\ell^c}^{(2)}\right]\right) \\ &+ \cos\left(g_{\mathrm{cls}}\left(\mathbf{c}_{\theta,\ell^c}^{(2)}\right), \mathrm{sg}\left[\mathbf{c}_{\theta',\ell^c}^{(1)}\right]\right)\Big),\end{aligned} \tag{9}$$

where g_{cls} is a separate two-layer student projection head.

Unlike the patch objective in Eq. 8, class-token alignment is cross-view: each student representation is matched to the EMA target from the other noise branch. The alignment therefore encourages the class token to capture information shared across independent observations rather than noise-specific details. This is the only additional objective introduced by our method; the flow and local patch objectives remain unchanged.

The patch and CLS objectives use separate two-layer student projection heads. The projection heads are shared across the two student views but not across objectives. Teacher targets are raw EMA features and receive no projection or gradient. The projection heads are discarded for generation and downstream frozen-feature evaluation.

3.3 Overall Objective and Controlled Comparison

The complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{flow}} + \lambda_{\text{patch}}\mathcal{L}_{\text{patch}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}}. \tag{10}$$

Here, λ_{patch} and λ_{cls} weight the patch and class-token alignment objectives, respectively. The teacher parameters are updated as an exponential moving average of the student:

$$\theta' \leftarrow \mu\theta' + (1-\mu)\theta. \tag{11}$$

where μ is the EMA decay coefficient.

Our baseline sets $\lambda_{\text{cls}} = 0$ but otherwise uses the same model, added CLS token, independently noised observations, flow loss, patch loss, and EMA teacher. Our method instead sets $\lambda_{\text{cls}} > 0$. This design isolates the effect of adding direct cross-view supervision to the CLS token from changes in architecture, corruption, or generative supervision.

The global and local objectives need not read from the same transformer block. We therefore use separate notation for the CLS depth ℓ^c, the student patch depth ℓ^p_s, and the teacher patch depth ℓ^p_t. Their dataset-specific values are given in Table 1.

Several prior works also pursue generative models with explicit representation capabilities. MAGE unifies image synthesis and representation learning through discrete masked-token modeling, while DDAE++ augments diffusion training with contrastive self-distillation and an explicit self-conditioning pathway; D^{3}CL instead contrastively adapts a pretrained Stable Diffusion model across noise levels (Li et al., 2023; Xiang et al., 2025; Dai et al., 2026). Our setting is complementary: we keep the generative and local patch objectives fixed, and the matched variants differ only in whether the added CLS token receives cross-view self-distillation. This isolates the effect of an explicit global semantic objective during flow pretraining, without changing the generative formulation or introducing a separate representation-feedback pathway.

4 Experiments

We evaluate whether cross-view class-token alignment strengthens semantic representations while maintaining generation quality. We first report the matched ImageNet comparison and text-to-image generation results, then analyze the learned representations and the choice of CLS alignment depth and loss weight.

4.1 Experimental Setup

ImageNet training. We train class-conditional SelfFlowDiT-XL/2 models on ImageNet-1K at 256×256 resolution (Deng et al., 2009). Images are encoded using the Stable Diffusion VAE, producing a 16×16 latent patch grid. Both the baseline and our method are trained for 1M steps and evaluated using EMA weights. Full optimizer, preprocessing, and sampling details are provided in Appendix A.

Text-to-image training. To test whether the objective extends beyond class-conditional generation, we apply it to a Flux-style text-conditioned diffusion transformer trained on 1M text–image

Table 1: **Training configurations.** Layer numbers are one-based transformer-block indices. The baseline uses the same settings with $\lambda_{\text{cls}} = 0$.

Setting	ImageNet	Text-to-image
Backbone	SelfFlowDiT-XL/2, 28 blocks	Flux-style DiT, 21 feature layers
Training data	ImageNet-1K	1M text–image pairs
Resolution	256×256	256×256
Reported training step	1M	400K
Global batch size	256	512
Dual-timestep mask ratio ρ	0.50	0.75
Patch student $\rightarrow$ teacher depth	$8 \rightarrow 20$	$6 \rightarrow 15$
CLS student $\rightarrow$ teacher depth	$18 \rightarrow 18$	$15 \rightarrow 15$
λ_{patch}	0.8	0.8
λ_{cls}	0.2	2.0
EMA decay μ	0.9999	0.9999

Table 2: **ImageNet representation and generation at 1M steps.** Patch linear probing and VOC segmentation use features from transformer block 20. Both models contain the same CLS token and differ only in whether its cross-view alignment objective is active.

Method	FID $\downarrow$	[CLS] linear $\uparrow$	Patch linear $\uparrow$	VOC mIoU $\uparrow$
Two-view SelfFlow	**5.12**	63.49	60.03	57.15
+ Class-token alignment	5.26	**72.90**	**70.11**	**60.79**
Difference	$+0.14$	$+9.41$	$+10.08$	$+3.64$

pairs. The CLS and patch objectives are applied only to the image stream; the text-conditioning pipeline is unchanged. We compare both variants at 400K training steps. Full architecture and evaluation details are provided in Appendix B.

Representation evaluation. We evaluate frozen global and patch representations with linear classifiers. For the global probe, we extract the raw CLS token from block 18. For the patch probe, we extract raw patch tokens from block 20 and average them spatially. For each probe, we standardize every feature dimension using training-set statistics before fitting the classifier. Following diffusion-representation evaluation (Xiang et al., 2023), features are extracted from noised latents at a fixed evaluation timestep, using the null class condition so that no ground-truth label is provided to the diffusion backbone.

For dense evaluation, we freeze the backbone and train a per-token linear segmentation head on Pascal VOC2012 (Everingham et al., 2010). The head applies LayerNorm followed by a linear classifier to the raw patch grid and bilinearly upsamples its predictions to the input resolution. We report mean intersection-over-union (mIoU). Probe optimization and hyperparameter search are described in Appendix C.

Generation evaluation. For ImageNet, we report FID from 50,000 EMA samples against the standard ImageNet-256 ADM reference (Heusel et al., 2017). For text-to-image generation, we generate one sample for each of 50,000 held-out prompts and compute FID against the corresponding validation images and CLIP score against the prompts (Radford et al., 2021). All text-to-image comparisons use the same sampler, reference set, feature extractor, and preprocessing. These FID values are intended for controlled comparison between our two variants rather than for direct comparison with results obtained using a different evaluator.

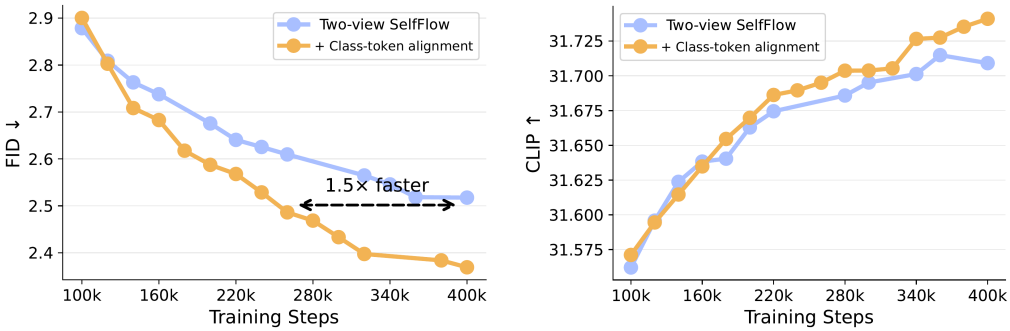
4.2 IMAGENET RESULTS

Table 2 shows that the additional class-token alignment objective substantially improves both global and local representations. Class-token linear probing increases by 9.41 points, from 63.49% to



(a) ImageNet FID. (b) Patch-token linear probe. (c) VOC2012 dense probe.

Figure 2: **ImageNet generation and representation quality throughout training.** We compare the matched two-view SelfFlow baseline with our model using cross-view class-token alignment. From left to right, we report ImageNet FID, patch-token linear-probe accuracy, and dense-probe mIoU across training checkpoints. Class-token alignment yields persistent improvements in both patch-level and dense representations while maintaining comparable generation quality.



(a) FID vs. training steps. (b) CLIP score vs. training steps.

Figure 3: **Text-to-image generation throughout training.** We compare the matched two-view SelfFlow baseline with our model using cross-view class-token alignment. FID (left) and CLIP score (right) are evaluated across training checkpoints using the same sampling and evaluation pipeline.

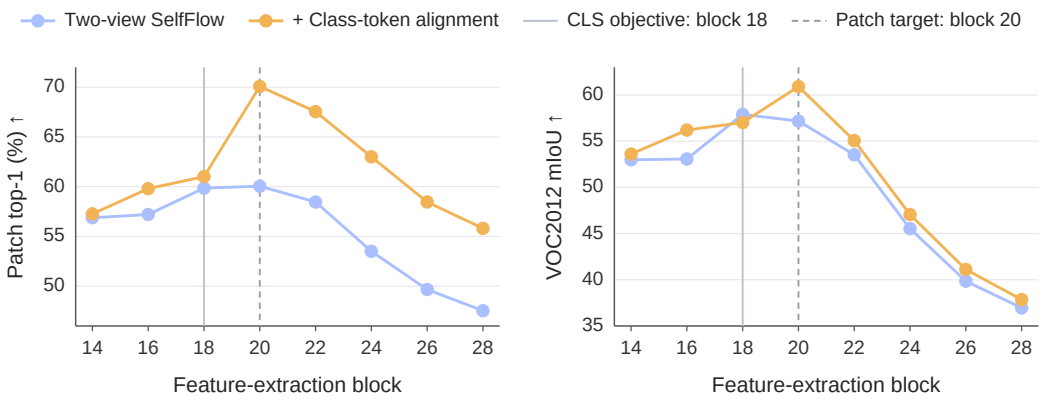
72.90%. Mean-patch linear probing improves by 10.08 points, from 60.03% to 70.11%, and frozen VOC2012 segmentation improves by 3.64 mIoU, from 57.15 to 60.79.

The patch and dense improvements cannot be explained by a better global readout alone: both evaluations operate directly on the frozen patch tokens. Although the only *additional* semantic objective is applied to the CLS token, it changes the representations learned by the image-token stream under the otherwise unchanged flow and patch objectives.

These representation gains are achieved with comparable ImageNet generation FID, which changes from 5.12 to 5.26 at the 1M-step checkpoint. Figure 2 further compares FID, patch-token linear probing, and VOC2012 mIoU across training checkpoints. The patch and dense representation gains develop during training, while generation FID follows a similar trajectory for the two models.

4.3 Text-to-Image Results

We next evaluate whether the same class-token alignment objective also benefits text-conditioned generation. The baseline and our model use the same architecture, training data, independently noised observations, patch objective, and generative objective. They differ only in whether cross-view class-token alignment is active.



(a) ImageNet mean-patch linear probe.

(b) VOC2012 frozen dense probe.

Figure 4: **Patch representation quality across transformer depth.** The vertical markers indicate the class-token alignment block (18) and the EMA patch-target block (20). The largest gains in both ImageNet patch linear probing and VOC2012 dense probing appear at block 20 rather than at the directly supervised CLS block.

At the matched 400K checkpoint, the additional objective reduces FID from 2.517 to 2.369, while improving CLIP score from 31.709 to 31.741. The training curves in Figure 3 also show faster FID convergence, while the differences in CLIP score remain small. Thus, the same objective that strengthens ImageNet representations also benefits generation FID in this text-to-image setting.

4.4 Representation Analysis

Feature-extraction depth. We first examine where the patch-level gains appear within the trained model. The 1M-step checkpoints and the CLS alignment block (18) are fixed; only the block used to extract patch features is varied. Figure 4 reports frozen patch evaluation across the later transformer blocks. At block 18, patch linear accuracy improves only from 59.85% to 61.01%, while VOC mIoU changes from 57.91 to 57.02. At block 20, however, patch linear accuracy increases from 60.05% to 70.08% and VOC mIoU increases from 57.15 to 60.79.

Block 20 is also the layer used as the EMA target for the inherited local patch objective. The strongest patch-level gains therefore appear at the patch-teacher depth rather than at the directly aligned CLS depth. Large patch-linear improvements remain visible in subsequent blocks.

CLS-to-patch attention. Figure 5 compares matched CLS-to-patch attention maps. In these selected examples, our method concentrates attention more strongly on class-relevant regions—the muzzle, matchsticks, animal heads, and crossword grid—than the baseline. These depth comparisons and attention maps describe the learned representations but do not establish a particular causal mechanism for the patch-level gains.

4.5 Ablation Studies

We select the ImageNet CLS attachment depth and loss weight using 100K-step pilot experiments, evaluating representation quality with weighted kNN and generation with FID (Figure 6).

CLS attachment depth. The response is non-monotonic: block 18 gives the highest same-block patch kNN accuracy, block 28 the highest CLS kNN accuracy, and block 20 the lowest FID. We choose block 18 as a trade-off among these criteria.

CLS objective weight. At block 18, CLS kNN gains begin to saturate around $\lambda_{\text{cls}} = 0.2$–$0.3$. Increasing the weight from 0.3 to 0.5 worsens FID without improving CLS kNN accuracy. We use $\lambda_{\text{cls}} = 0.2$, which has lower FID than 0.3 with slightly lower CLS and patch accuracies. Exact results are provided in Tables 7 and 8 in Appendix D.

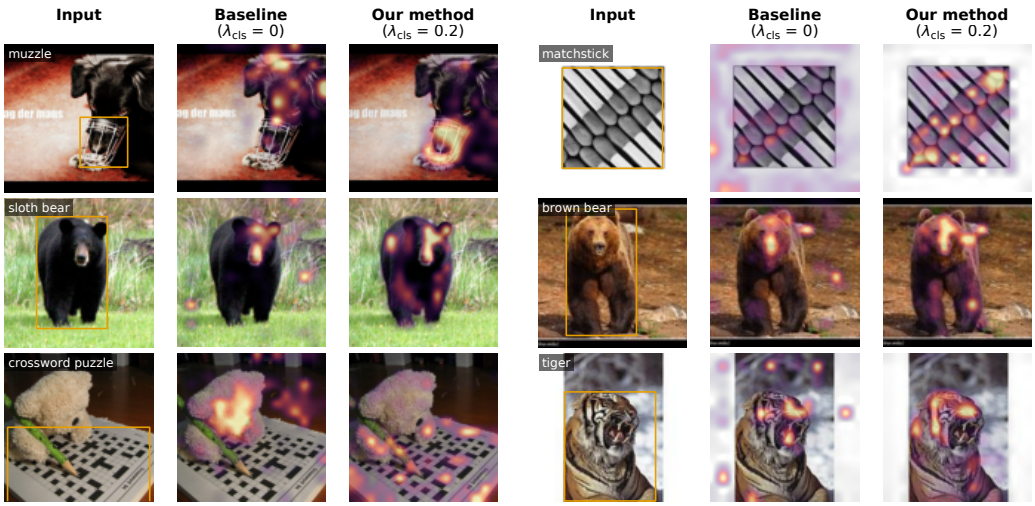


Figure 5: **CLS-to-patch attention maps on ImageNet.** Orange boxes indicate the target object regions in the input images. We compare the matched baseline ($\lambda_{\text{cls}} = 0$) with our model using class-token alignment ($\lambda_{\text{cls}} = 0.2$). Across these selected examples, the aligned model concentrates attention more strongly on class-relevant and discriminative image regions.

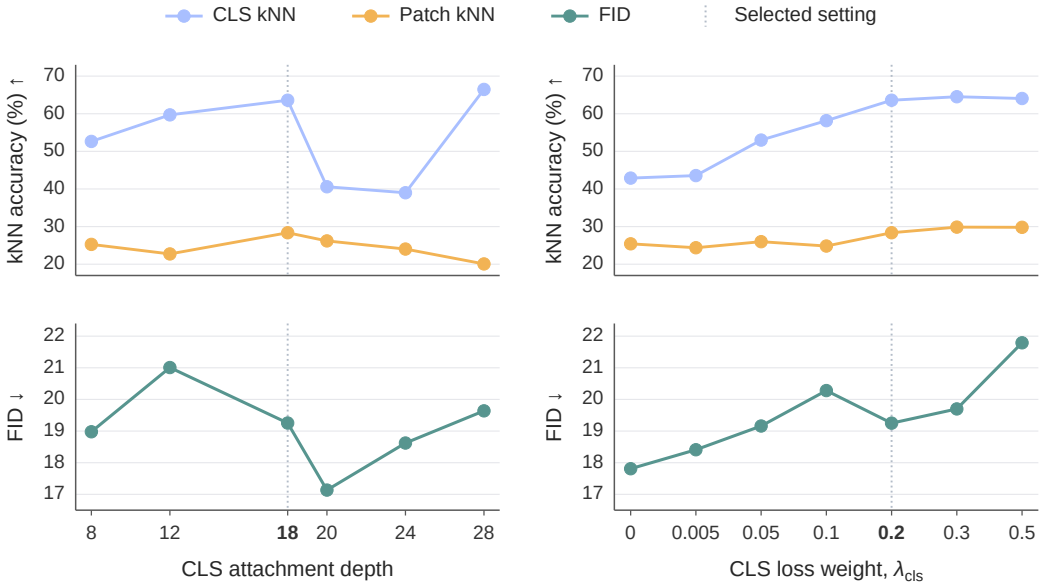


Figure 6: **CLS-objective ablations at 100K ImageNet steps.** The attachment-depth sweep evaluates weighted kNN at the corresponding attachment block; the loss-weight sweep fixes the attachment at block 18. Top: CLS and patch kNN accuracy. Bottom: FID. Dotted guides and bold ticks mark the selected settings (block 18 and $\lambda_{\text{cls}} = 0.2$). Exact values are in Tables 7 and 8.

5 Conclusion

We studied whether semantic representation can be explicitly strengthened during diffusion pretraining without sacrificing generation. Starting from a matched two-view SelfFlow baseline, we add cross-view class-token alignment while retaining the same generative and local patch objectives. This simple addition substantially improves global and patch-token probing and dense semantic segmentation, while preserving ImageNet generation and improving FID and CLIP score in text-to-image training. These results suggest that representation need not remain a by-product of generation or merely a tool for improving it; it can be optimized directly alongside synthesis as a first-class outcome of diffusion pretraining.

AI USE STATEMENT

Generative AI tools, including OpenAI Codex, were used to assist with drafting and language editing, manuscript organization, literature discovery and summarization, citation checking and formatting, and critical review of the manuscript, including feedback on the presentation and interpretation of previously obtained experimental results. The authors reviewed all AI-assisted text and suggestions and checked cited references against the original sources. The authors take responsibility for the final content of the manuscript, including text and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 3 describes the two-view training framework and cross-view class-token alignment objective. Appendices A and B provide architecture, optimization, and generation-evaluation details. Appendix C specifies representation-evaluation protocols and hyperparameter searches, while Appendix D reports detailed ablation results. Our anonymized code repository is available at `https://anonymous.4open.science/r/SiT-IBOT-7E33`.

REFERENCES

Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=SlxSY2UZQT`.

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*, pp. 9650–9660, 2021.

Hila Chefer, Patrick Esser, Dominik Lorenz, Dustin Podell, Vikash Raja, Vinh Tong, Antonio Torralba, and Robin Rombach. Self-supervised flow matching for scalable multi-modal synthesis. *arXiv preprint arXiv:2603.06507*, 2026.

Yasong Dai, Zeeshan Hayder, David Ahmedt-Aristizabal, and Hongdong Li. Probing diffusion denoising dynamics for contrastive representation learning. *arXiv preprint arXiv:2607.09067*, 2026.

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*, pp. 248–255. Ieee, 2009.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010.

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.

Dengyang Jiang, Mengmeng Wang, Liuzhuozheng Li, Lei Zhang, Haoyu Wang, Wei Wei, Guang Dai, Yanning Zhang, and Jingdong Wang. Representation alignment for diffusion transformers without external components. In Carl Vondrick, Bharath Hariharan, Colin Raffel, Lerrel Pinto, Diyi Yang, and Aleksandra Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 18884–18915, 2026.

Theodoros Kouzelis, Efstathios Karypidis, Ioannis Kakogeorgiou, Spyridon Gidaris, and Nikos Komodakis. Boosting generative image modeling via joint image-feature synthesis. In Danielle Belgrave, Cheng Zhang, Haoping Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, and Nan Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 16685–16714. Curran Associates, Inc., 2025. doi: 10.52202/085713-0563. URL `https://proceedings.neurips.cc/paper_files/paper/2025/file/186a213d720568b31f9b59c085a23e5a-Paper-Conference.pdf`.

Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. Mage: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2142–2152, 2023.

Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 36:47500–47510, 2023.

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*, pp. 8748–8763. PmLR, 2021.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in neural information processing systems*, 36: 1363–1389, 2023.

Chen Wei, Karttikeya Mangalam, Po-Yao Huang, Yanghao Li, Haoqi Fan, Hu Xu, Huiyu Wang, Cihang Xie, Alan Yuille, and Christoph Feichtenhofer. Diffusion models as masked autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16284–16294, 2023.

Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, Ming-Ming Cheng, and Xiang Li. Representation entanglement for generation: Training diffusion transformers is much easier than you think. In Danielle Belgrave, Cheng Zhang, Haoping Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, and Nan Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 7714–7743. Curran Associates, Inc., 2025. doi: 10.52202/085713-0264. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/0b99315234cc95e6ef281f9155b68832-Paper-Conference.pdf.

Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15802–15812, 2023.

Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Ddae++: enhancing diffusion models towards unified generative and discriminative learning. *arXiv preprint arXiv:2505.10999*, 2025.

Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Conditioning residuals for diffusion models via representation feedback, 2026. URL https://arxiv.org/abs/2505.10999.

Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18938–18949, 2023.

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. In Yisong Yue, Abhinav Garg, Nan Peng, Fei Sha, and Rose Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 87400–87442, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d9e42b4d7163931f3689d6d6fbaa11d0-Paper-Conference.pdf.

Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5729–5739, 2023.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In Carl Vondrick, Bharath Hariharan, Colin Raffel, Lerrel Pinto, Diyi Yang, and Aleksandra Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 35791–35820, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/3c4141c12660ad3625eb4ae845e0a6f9-Paper-Conference.pdf.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8435–8445, June 2024.

A IMAGENET TRAINING AND GENERATION DETAILS

Architecture and latent encoding. We use a class-conditional SelfFlowDiT-XL/2 backbone with the configuration in Table 3, initialized from scratch. Images are resized and center-cropped to 256×256, horizontally flipped with probability 0.5 during pretraining, and encoded using `stabilityai/sd-vae-ft-ema`. Pretraining samples the VAE posterior and scales the latent by 0.18215; frozen-feature extraction instead uses the posterior mode with the same scaling. Each $32 \times 32 \times 4$ latent is divided into non-overlapping 2×2 patches, giving a 16×16 image-token grid. Both variants prepend a learnable CLS token.

Training and alignment heads. Both models are trained for 1M optimizer steps and evaluated using EMA weights. The global batch contains 256 source images before two-view expansion, yielding 512 student-view instances per update. Classifier-free conditioning dropout is sampled independently for the two student forwards; both EMA-teacher forwards retain the original class labels. The dropout probability and optimizer settings are listed in Table 3.

Each active student projection head is $1152 \to 2304 \to 1152$, with SiLU between the linear layers. Our method uses separate patch and CLS heads; the baseline retains the CLS token but does not enable the CLS projection head or objective. Teacher targets are raw, stop-gradient EMA features. Projection heads are discarded for generation and frozen-feature evaluation.

For CLS conditioning, we zero the timestep-embedding contribution before adding class conditioning. This is not equivalent to passing numerical timestep zero through the timestep embedder. Both ImageNet variants use this convention. Token masks are sampled independently from a Bernoulli distribution, with cleaner-token probability $\rho = 0.5$ as defined in the method.

Table 3: **ImageNet training configuration.** Transformer-block indices are one-based. The baseline uses the listed backbone and optimization settings with $\lambda_{\text{cls}} = 0$ and no active CLS projection head.

Setting	Value
Dataset	ImageNet-1K
Resolution	256×256
Backbone	SelfFlowDiT-XL/2
Transformer blocks	28
Hidden dimension	1152
Attention heads	16
Latent patch grid	16×16
Global batch (source images)	256
Optimizer	AdamW
Learning rate	10^{-4}, constant
Adam betas	$(0.9, 0.999)$
Weight decay / warmup steps	0 / 0
Training steps	1M
EMA decay	0.9999
Student class-dropout probability	0.1
Cleaner-token probability ρ	0.5
Patch student $\to$ teacher block	$8 \to 20$
CLS student/teacher block	18
λ_{patch}	0.8
λ_{cls}	0.2

Generation evaluation. We compute FID (Heusel et al., 2017) from 50,000 EMA samples using the ADM/guided-diffusion evaluator and its labeled ImageNet-256 reference, `VIRTUAL_imagenet256_labeled.npz`. Sampling records identify the adaptive Dopri5 ODE solver with a nominal count of 250. The solver API interprets this parameter as requested output times, not a verified number of function evaluations. Classifier-free guidance uses

$$v_s = v_{\text{uncond}} + s\big(v_{\text{cond}} - v_{\text{uncond}}\big), \qquad s = 1, \tag{12}$$

so the evaluated setting returns the conditional velocity prediction. Both variants use the same recorded sampling and evaluation configuration.

B TEXT-TO-IMAGE TRAINING AND EVALUATION DETAILS

Data and split construction. We use the dataset `nyu-visionx/scale-rae-data`, specifically its `flux_training_2_47m` subset. Shards are processed sequentially, using seed 42 and a nominal validation-assignment probability of $1/21$, subject to quotas of 1M training and 50K validation pairs. The resulting manifests have no overlapping source IDs or prompts under either exact or whitespace-normalized matching. This check does not test for duplicates in image content.

Architecture and conditioning. We train a Flux-style diffusion transformer from scratch at 256×256 resolution. Its seven double-stream and fourteen single-stream blocks provide 21 feature layers. The frozen VAE is `stabilityai/sd-vae-ft-mse`. A frozen Qwen3-4B text-encoder preset concatenates features from its configured layers $\{9, 18, 27\}$ to form 7680-dimensional conditioning features. Prompt dropout is applied with probability 0.1, and the resulting context is shared across both student and both EMA-teacher forwards.

Optimization and objectives. The shared data and optimization settings are summarized in Table 4. A global batch contains $8 \times 32 \times 2 = 512$ source images across eight ranks and two gradient accumulation steps, before two-view expansion. The patch objective matches image-stream features from student block 6 to teacher block 15. Our method additionally applies cross-view CLS alignment at block 15. Separate runs explored $\lambda_{\text{cls}} \in \{0.5, 1, 2, 4\}$; the reported 400K-step comparison uses $\lambda_{\text{cls}} = 2$.

The text-to-image mask independently selects the noisier branch with probability 0.25. Under the method's convention that ρ denotes cleaner-token probability, this corresponds to $\rho = 0.75$.

Table 4: **Text-to-image training configuration.** Diffusion-transformer block indices are one-based. The table lists shared architecture and optimization settings, with the method's CLS objective; the baseline uses $\lambda_{\text{cls}} = 0$. Here ρ denotes cleaner-token probability.

Setting	Value
Training / validation pairs	1M / 50K
Resolution	256×256
Hidden dimension	1152
Attention heads	16
Double-stream / single-stream blocks	7 / 14
Text-conditioning dimension	7680
Reported training checkpoint	400K
Global batch (source images)	512
Ranks $\times$ per-rank batch $\times$ accumulation	$8 \times 32 \times 2$
Optimizer	AdamW
Learning rate	10^{-4}, constant
Adam betas	$(0.9, 0.999)$
Weight decay / warmup steps	$0 \,/\, 0$
Gradient clipping	1.0
EMA decay	0.9999
Prompt-dropout probability	0.1
Cleaner-token probability ρ	0.75
Patch student $\rightarrow$ teacher block	$6 \rightarrow 15$
CLS student/teacher block	15
λ_{patch}	0.8
λ_{cls}	2.0

Sampling and FID. We generate one image for each of 50,000 held-out prompts using the EMA checkpoint at 256×256 resolution. The reported evaluation settings are 50 sampling steps and guidance scale 1.0. FID compares generated images with the corresponding 50,000 validation images.

The text-to-image evaluator uses `torchvision` Inception V3 with `DEFAULT` pretrained weights, `fc=Identity`, and `transform_input=False`. Images are converted to RGB and bicubically resized to 256×256. The weights' preprocessing then resizes with bilinear interpolation to size 342 and center-crops to 299×299 before feature extraction. We use this same pipeline and reference set for every reported text-to-image FID. These values are intended for within-pipeline comparisons, not direct comparison with FID values from the separate ADM evaluation used for ImageNet or other feature pipelines.

CLIP score. We use `openai/clip-vit-base-patch32` (Radford et al., 2021). For image and text embeddings independently normalized to unit length, $\hat{e}_i^{\text{image}}$ and $\hat{e}_i^{\text{text}}$, the score is

$$\text{CLIP} = \frac{100}{n} \sum_{i=1}^{n} \max\left\{ \left(\hat{e}_i^{\text{image}}\right)^{\mathsf{T}} \hat{e}_i^{\text{text}},\, 0 \right\}, \qquad n = 50{,}000. \tag{13}$$

Thus, negative cosine similarities are clamped to zero before averaging. All comparisons use the same prompts and CLIP preprocessing.

C Representation Evaluation

C.1 ImageNet Linear Probing

Feature extraction. We evaluate the frozen 1M-step EMA checkpoints on all 1,281,167 ImageNet training images and all 50,000 validation images. Images are encoded using the deterministic VAE posterior mode with the 0.18215 scaling described in Appendix A. At evaluation timestep $t = 0.8$, the backbone receives

$$z_t = 0.8z + 0.2\epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \tag{14}$$

consistent with the transport convention in the method. The backbone receives the null class condition, not the ground-truth ImageNet label.

For the global representation, we extract the raw CLS token after block 18. For the local representation, we spatially average the raw 16×16 patch-token grid after block 20. We do not apply samplewise ℓ_2 normalization. Instead, each feature dimension is standardized using its mean and standard deviation on the training set; the same statistics are applied to validation features.

Each image receives one Gaussian draw during extraction, and its resulting feature is cached. These features are reused across probe epochs and learning-rate candidates without resampling diffusion noise. Within each readout, the two models use extraction seed 0 and matching execution settings: three GPU ranks for CLS features and four for mean-patch features. Noise tensors were not retained, so exact per-image matching between models has not been independently verified. CLS and mean-patch caches are extracted separately and are not assumed to share noise realizations.

Probe training and selection. For every model and readout, we train a single affine classifier using SGD with momentum 0.9, zero weight decay, batch size 8192, and 60 epochs. The learning rate is constant, with no warmup. All four final probes use the same 33-value learning-rate grid:

$$\begin{aligned}\mathcal{H}_{\text{linear}} = \{&0.005,\ 0.0075,\ 0.010,\ 0.0125,\ 0.015,\ 0.0175,\ 0.020,\ 0.0225,\\ &0.025,\ 0.030,\ 0.040,\ 0.050,\ 0.060,\ 0.070,\ 0.080,\ 0.090,\\ &0.100,\ 0.120,\ 0.140,\ 0.160,\ 0.180,\ 0.200,\ 0.240,\ 0.280,\\ &0.320,\ 0.360,\ 0.420,\ 0.500,\ 0.600,\ 0.750,\ 0.900,\ 1.100,\ 1.300\}.\end{aligned}$$

We independently select the learning rate and epoch that maximize validation top-1 accuracy for each model and readout. Thus, the search space and selection rule are identical, but the selected hyperparameters need not be. Table 5 gives the selected settings and final results.

Table 5: **Selected settings for final ImageNet linear probes.** Each probe uses the same 33-rate search and 60-epoch training budget. Bold and underline mark the higher and lower accuracy within each readout, respectively.

Model	Learning rate	Epoch	Top-1 (%)
CLS, block 18			
Baseline	0.12	54	<u>63.488</u>
Ours	0.10	17	**72.898**
Mean patch, block 20			
Baseline	0.18	58	<u>60.03</u>
Ours	0.12	57	**70.11**

Evaluation sweeps. The feature-extraction-depth and checkpoint-trend patch probes use 27-rate and 29-rate grids, respectively. Relative to the final grid above,

$$\mathcal{H}_{\text{depth}} = \mathcal{H}_{\text{linear}} \setminus \{0.005,\ 0.0075,\ 0.010,\ 0.0125,\ 0.0175,\ 0.0225\},$$
$$\mathcal{H}_{\text{trend}} = \mathcal{H}_{\text{depth}} \cup \{1.600,\ 2.000\}.$$

Both sweeps use the same 60-epoch budget, batch size 8192, SGD momentum, zero weight decay, constant learning rates without warmup, training-set standardization, and best-validation selection described above. At 1M steps, the depth, trend, and final mean-patch evaluations reuse the same feature caches; differences between these evaluation series are therefore not independent diffusion-noise trials. The block-20 patch readout was chosen from the blockwise evaluation, whose full curve is also reported.

C.2 Pascal VOC2012 Dense Probing

Data and preprocessing. We evaluate frozen dense representations on Pascal VOC2012 (Everingham et al., 2010), using 1,464 training images and 1,449 validation images. When needed, images undergo progressive BOX downsampling, followed by bicubic resizing of the shorter side to 256 and a 256×256 center crop. Segmentation masks undergo the same geometric transformations with nearest-neighbor interpolation at every resize, preserving class IDs and the void label 255.

Features and dense head. The frozen backbone receives null ImageNet class ID 1000 during both training and validation. VOC labels supervise only the segmentation head. The head applies LayerNorm and a tokenwise linear classifier over 21 classes to the raw 16×16 patch-token grid from block 20. Logits are bilinearly upsampled to the mask resolution with `align_corners=False`. We optimize cross-entropy with ignore index 255 and report mIoU from the 21-class confusion matrix.

Features use $t = 0.8$ under Eq. 14. Unlike ImageNet linear probing, this evaluation draws fresh Gaussian noise on every feature-extraction call, including each validation pass. All learning-rate heads share the features from a given batch. We do not average predictions across multiple noise realizations.

Optimization and selection. We use Adam with betas $(0.9, 0.999)$, zero weight decay, batch size 32, and cosine learning-rate annealing over 60 epochs without warmup. The initial learning-rate grid is

$$\mathcal{H}_{\text{VOC}} = \{0.0002,\ 0.0003,\ 0.0005,\ 0.00075,\ 0.001,\ 0.0015,\ 0.002,\ 0.003,\ 0.0045,\ 0.006\}.$$

Each model independently selects the learning rate and epoch with the highest validation mIoU. Both select an initial learning rate of 0.0045: the baseline reaches 57.15 mIoU at epoch 48, and our method reaches 60.79 mIoU at epoch 19.

C.3 WEIGHTED kNN EVALUATION

The 100K-step CLS ablations use weighted kNN rather than a trained linear probe. We construct a labeled feature bank from all 1,281,167 ImageNet training images and evaluate all 50,000 validation images. Frozen EMA features are extracted using a 256×256 center crop, null conditioning, $t = 0.8$, and one Gaussian noise draw per image. Extraction uses seed 0, four GPU ranks, and batch size 128 per rank.

We ℓ_2-normalize the CLS vector or mean-pooled patch vector and retrieve the $k = 20$ nearest neighbors by normalized inner product. For a normalized query $\hat{q}$, normalized bank features $\hat{f}_i$, and bank labels y_i, the prediction is

$$\hat{y}(\hat{q}) = \arg\max_{c} \sum_{i\in\mathcal{N}_{20}(\hat{q})} \mathbf{1}\{y_i = c\} \exp\left(\frac{\hat{q}^{\mathsf{T}}\hat{f}_i}{0.07}\right), \tag{15}$$

where $\mathcal{N}_{20}(\hat{q})$ is the retrieved neighbor set. The same k and temperature are used throughout both ablations. The attachment-depth sweep reads features at each model's attachment block; the loss-weight sweep always reads block 18.

C.4 SINGLE-BLOCK AND MULTI-BLOCK FEATURES

We additionally concatenate features from blocks $\{14, 16, 18, 20\}$. For ImageNet, we mean-pool each block separately, concatenate the four 1152-dimensional vectors, and standardize each dimension of the resulting 4608-dimensional vector using training-set statistics. A single affine classifier is trained with the 27-rate grid $\mathcal{H}_{\text{depth}}$ and the optimization protocol in Appendix C.1. The selected learning rates and epochs are 0.07/60 for the baseline and 0.06/56 for our method.

For VOC, block features are concatenated along the raw token-channel dimension. The head is one $\mathrm{LayerNorm}(4608)$ followed by $\mathrm{Linear}(4608, 21)$. Both models select learning rate 0.003, with the baseline and our method selecting epochs 52 and 48, respectively. The resulting scores are listed in Table 6. Concatenation improves the reported absolute VOC mIoU for both models, while the between-model gain is larger at block 20 alone.

The single-block ImageNet row retains the auxiliary scores reported with this comparison; the final 33-rate results are listed separately in Table 5.

Table 6: **Single-block and multi-block patch evaluations.** ImageNet reports linear-probe top-1 accuracy; VOC reports dense-probe mIoU. Δ is the difference between ours and baseline, in percentage points. Within each dataset, bold and underline mark the highest and second-highest value in each numerical column.

Feature blocks	Baseline	Ours	Δ
ImageNet			
20	<u>60.052</u>	<u>70.084</u>	**+10.032**
$\{14, 16, 18, 20\}$	**62.602**	**70.516**	<u>+7.914</u>
VOC2012			
20	<u>57.180</u>	<u>60.912</u>	**+3.732**
$\{14, 16, 18, 20\}$	**60.399**	**63.104**	<u>+2.705</u>

D NUMERICAL RESULTS FOR CLS ABLATIONS

Tables 7 and 8 provide the exact values for the main-text CLS ablation figure. All checkpoints are evaluated at 100K training steps using the weighted kNN protocol in Appendix C.3. The depth sweep varies both the loss attachment and representation readout block, rather than isolating attachment depth under a fixed readout. The weight sweep fixes both at block 18; fixed objective weights are given in the table captions. The main ImageNet configuration uses block 18 and $\lambda_{\text{cls}} = 0.2$ as a trade-off, not as the optimum for every metric. Bold and underline mark the best and second-best value in each metric column.

Table 7: **CLS attachment-depth ablation at 100K steps.** $\lambda_{\text{cls}} = 0.2$ and $\lambda_{\text{patch}} = 0.8$ throughout. Representation metrics use features from the corresponding attachment block. kNN accuracies are percentages.

CLS block	FID ↓	CLS kNN ↑	Patch kNN ↑
8	18.978	52.638	25.268
12	21.008	59.696	22.732
18	19.254	<u>63.592</u>	**28.366**
20	**17.135**	40.588	<u>26.194</u>
24	<u>18.622</u>	38.996	24.030
28	19.639	**66.458**	20.076

Table 8: **CLS-loss-weight ablation at block 18 and 100K steps.** $\lambda_{\text{patch}} = 0.8$ throughout. kNN accuracies are percentages.

λ_{cls}	FID ↓	CLS kNN ↑	Patch kNN ↑
0.000	**17.81**	42.89	25.42
0.005	<u>18.41</u>	43.56	24.38
0.050	19.16	53.00	25.99
0.100	20.28	58.16	24.83
0.200	19.25	63.59	28.37
0.300	19.70	**64.52**	**29.86**
0.500	21.79	<u>64.05</u>	<u>29.81</u>

E QUALITATIVE TEXT-TO-IMAGE EXAMPLES

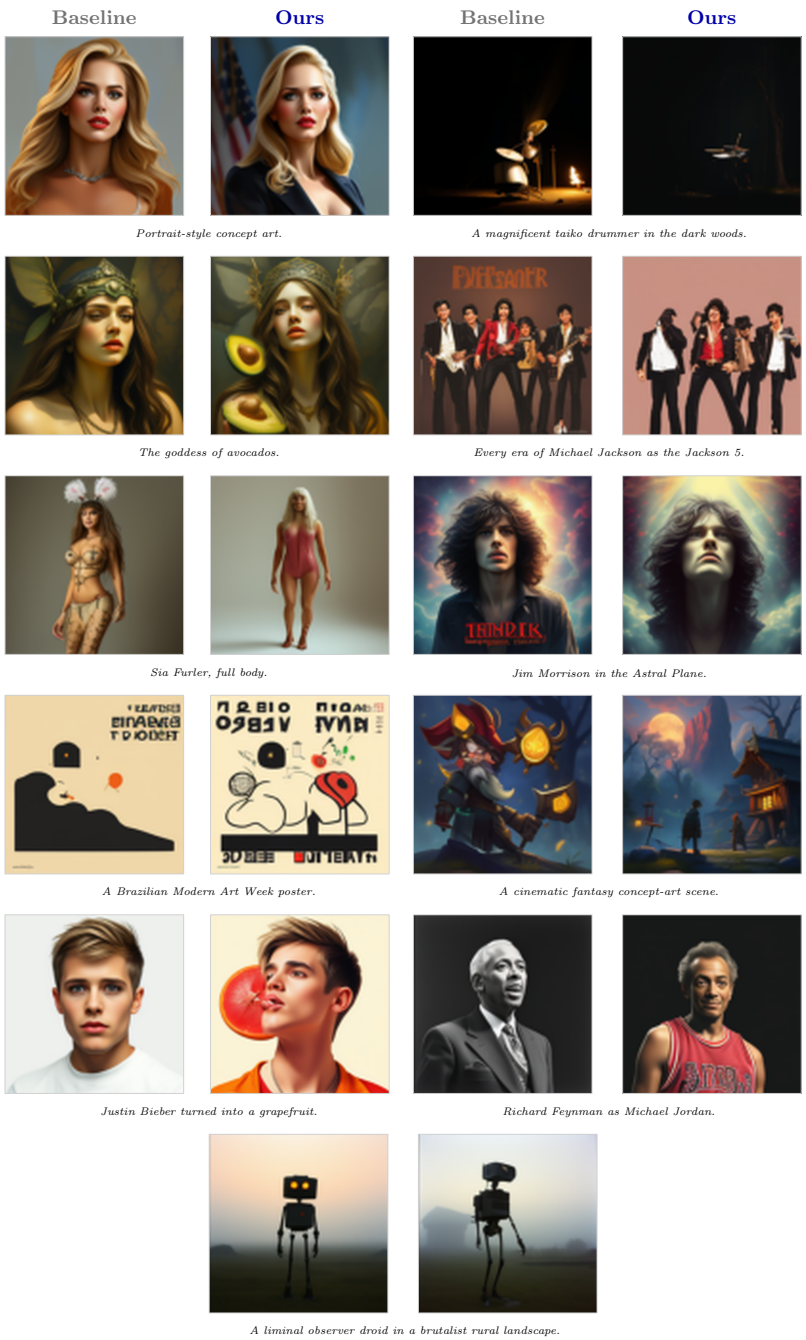


Figure 7: **Selected text-to-image comparisons at 400K steps.** Within each pair, both models use the same prompt, random seed, and sampling configuration. Examples were selected post hoc according to per-sample CLIP improvement and illustrate individual cases rather than average performance. One pair, with prompt ID 3, is omitted because the baseline output is nearly blank. Displayed prompt labels are abbreviated; full prompts and seeds are listed in Table 9.

Table 9: **Full prompts and random seeds for Figure 7.** Original prompt IDs are retained.

ID	Exact prompt	Seed
1	Draw Lindsey Pelas as Gillian Anderson, the president of the United States, digital painting, ArtStation concept art, sharp-focus illustration art by Artgerm, H 704.	20415
2	I want an image of a magnificent taiko drummer in the dark woods.	17146
4	Make a picture of the goddess of avocados, by Donato Giancola.	49143
5	Produce an image depicting every era of Michael Jackson represented as all the members of the Jackson 5.	40777
6	Please visualize Sia Furler full body.	42874
7	I want an image of a festival poster of Jim Morrison in the Astral Plane, haunting digital art.	49751
8	Show the Brazilian Modern Art Week poster. The poster style is modernism and the details are minimal. The poster features a sequence of images, ideas, emotions, and sensations that usually occur involuntarily in the mind during certain stages of sleep. The background is light beige, modernism, cubism, minimalism, designed by George Condo as an image.	22867
9	Produce an image depicting photo cartoon illustration, digital painting, volumetric lighting by Feng Zhu, 3D, Alejandro Alvarez Alena, Aenami artworks in 4K, Beeple, by Carel Willink and Gregory Crewdson, by Thomas Kinkade, Hearthstone, League of Legends, Dofus, Overwatch, concept sheet.	34805
10	I'd like to see an image of Justin Bieber turned into a grapefruit.	20273
11	Create a picture that shows Richard Feynman as Michael Jordan.	20203
12	Generate imagery of the lanky liminal observer droid by Dennis Mejillones, in a brutalist yet rural landscape by Simon Stalenhag, 35 mm film photography, dawn, eerie fog.	46176

F LIMITATIONS

Given the computational cost of diffusion pretraining, we prioritize matched ImageNet comparisons and targeted ablations, with one pretraining run per configuration in the main experiments. Evaluating variability across pretraining seeds and extending the study to larger training scales remain directions for future work.

Our representation evaluation focuses on class-conditional ImageNet pretraining and dense transfer to Pascal VOC2012. Additional downstream tasks and frozen-feature evaluation of the text-to-image models would help assess the generality of the representation gains. Text-to-image improvements are metric-dependent, with mixed results on compositional benchmarks. Finally, our depth analyses and attention visualizations characterize the learned representations; further investigation could clarify how global alignment strengthens patch features.