AFFINETOK: SEMANTIC AFFINE CONSISTENCY FOR DIFFUSION-FRIENDLY VISUAL TOKENIZER

Anonymous authors
Paper under double-blind review

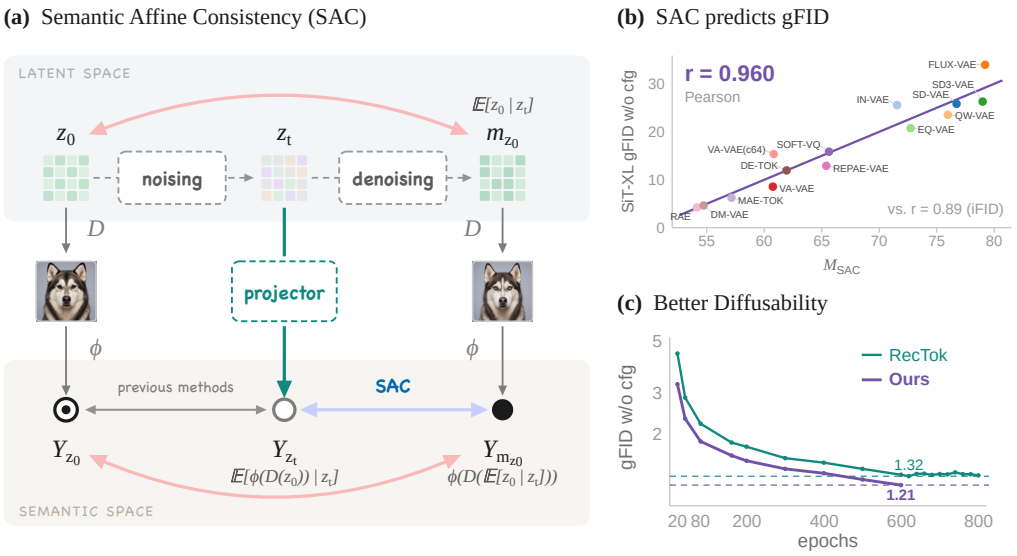

Figure 1: **Semantic Affine Consistency for diffusion-friendly visual tokenizers.** **(a)** Existing methods predict semantics directly from a noisy latent, which are the *average of clean-image semantics*. Diffusion instead predicts the posterior-mean clean latent and then decodes its semantics, which are the *semantics of the averaged clean latent*. SAC requires these two predictions to agree, a condition absent from prior supervision. **(b)** Our tokenizer-side metric $\mathcal{M}_{\mathrm{SAC}}$ strongly correlates with SiT-XL gFID across 14 visual tokenizers, reaching a Pearson correlation of 0.960. **(c)** SAC-guided training improves diffusability, reaching 1.21 gFID at 600 epochs without CFG.

ABSTRACT

Visual tokenizers increasingly inject semantic supervision into latent spaces to make downstream diffusion easier. Yet how these semantics should be organized to facilitate denoising remains underexplored. In this paper, we define the semantic recovery objective: the denoising process should recover the semantic content of the ground-truth image from noisy latent, and a good tokenizer should make it easier. Existing approaches train a projector to predict the semantics directly from the noisy latent. We argue that this predicts the *average of clean-image semantics*, whereas what really needs to be aligned is the *semantic of averaged clean latents*. More importantly, we demonstrate that the semantic recovery error *orthogonally* decomposes into the error of the optimal semantic prediction directly from the noisy latent and the error between these two semantic predictions. We therefore identify their consistency as the missing requirement and call it *Semantic Affine Consistency* (SAC). To examine whether this overlooked requirement is closely related to downstream generation, we introduce $\mathcal{M}_{\mathrm{SAC}}$, a tokenizer-side proxy for SAC. Across the evaluated tokenizers and diffusion model scales, $\mathcal{M}_{\mathrm{SAC}}$ closely tracks generation quality, reaching a Pearson correlation of 0.960 with SiT-XL gFID, thereby establishing SAC as a practical diagnostic and motivating SAC-guided tokenizer training. We then introduce AffineTok, which promotes SAC through two complementary components for tokenizer training. *Global Semantic*

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.

Coordination Token (GSCT) coordinates the semantic organization of clean latents, keeping semantic averaging meaningful, while *Posterior-Mean Semantic Alignment* (PMSA) predicts posterior-mean latents from noisy inputs and supervises their semantics. On ImageNet 256, compared with the baseline, AffineTok reduces gFID by 26% at 20 epochs and, with continued training, achieves a new state-of-the-art gFID of 1.21 without classifier-free guidance and 1.10 with guidance.

1 INTRODUCTION

In latent diffusion models, a visual tokenizer is more than a compressor: it determines the geometry of the denoising problem that the model must learn. This realization has driven a recent shift from tokenizers optimized primarily for reconstruction toward latent spaces explicitly shaped for generation. In particular, modern approaches inject features from visual foundation models (VFMs) into clean latents (Yao et al., 2025), regularize their structure (Kouzelis et al., 2025; Liu et al., 2025), or extend semantic supervision to noisy latent states (Shi et al., 2026b; Page et al., 2026). These methods share an appealing premise: if semantics are more accessible in the latent space, diffusion should be easier. Yet in diffusion, the semantic outcome is determined by decoding the predicted clean latent, rather than by directly reading the noisy state. This distinction raises a central question: *how should a tokenizer organize semantics in the latent space to make diffusion easier to learn?*

To answer this question, our starting point is the goal of denoising: to recover the clean image from a noisy observation, which involves recovering not only its latent representation but also its semantic content. Achieving this goal requires aligning clean-image semantics with those decoded from the predicted clean latent, with the latter corresponding to the *semantics of the averaged clean latent*. As shown in Fig. 1(a), existing methods instead apply semantic supervision through a different path, training a semantic projector to predict the clean-image semantics directly from the noisy latent. Under squared loss, the optimal projector predicts the *average of clean-image semantics*. As semantic mapping is nonlinear, averaging and semantic mapping need not commute, so current supervision does not ensure the required alignment of the decoded semantics. In fact, it can be exactly proven the semantic error of the image decoded from diffusion's clean-latent prediction decomposes orthogonally into two terms: the error of predicting semantics directly from the noisy latent and the gap between the average of semantics and the semantics of the averaged latent. Closing this gap requires the two semantic predictions to be consistent. We call this agreement *Semantic Affine Consistency* (SAC).

After identifying SAC as a requirement overlooked by existing tokenizer methods, we next ask whether tokenizers that better satisfy SAC support stronger downstream generation. Directly evaluating SAC would require training a downstream diffusion model for each tokenizer. Inspired by iFID (Xu et al., 2026), we instead construct probe points by interpolating between neighboring clean latents, using them as local proxies for posterior averaging. The resulting tokenizer-side probe metric, $\mathcal{M}_{\mathrm{SAC}}$, measures SAC violations at these points. Across the 14 visual tokenizers and diffusion model scales, $\mathcal{M}_{\mathrm{SAC}}$ closely tracks downstream generation quality (Fig. 1(b)), reaching a Pearson correlation of 0.960 with SiT-XL gFID.

SAC also guides tokenizer learning. Rather than enforcing SAC only at these probe points, which provide limited coverage of the latent space, we bring posterior-mean semantics under supervision through two complementary components. The *Global Semantic Coordination Token* (GSCT) uses a VFM-supervised global token to coordinate the organization of clean patch latents across images, keeping cross-image semantic averaging meaningful. The *Posterior-Mean Semantic Alignment* (PMSA) uses a lightweight predictor to estimate posterior-mean latents from noisy inputs and a semantic projector calibrated on clean latents to supervise their semantics. Both components are used during tokenizer training, while downstream diffusion training remains unchanged As shown in Fig. 1(c), on ImageNet 256×256, the resulting tokenizer improves gFID throughout diffusion training and ultimately reaches 1.21 without classifier-free guidance (cfg) and 1.10 with cfg.

Our contributions are as follows:

- We identify a previously overlooked mismatch between the semantics directly predicted from a noisy latent and those decoded from its posterior-mean clean-latent prediction. We formalize their agreement as SAC and derive an exact orthogonal decomposition showing their disagreement corresponds to a previously unaddressed term of semantic recovery error.

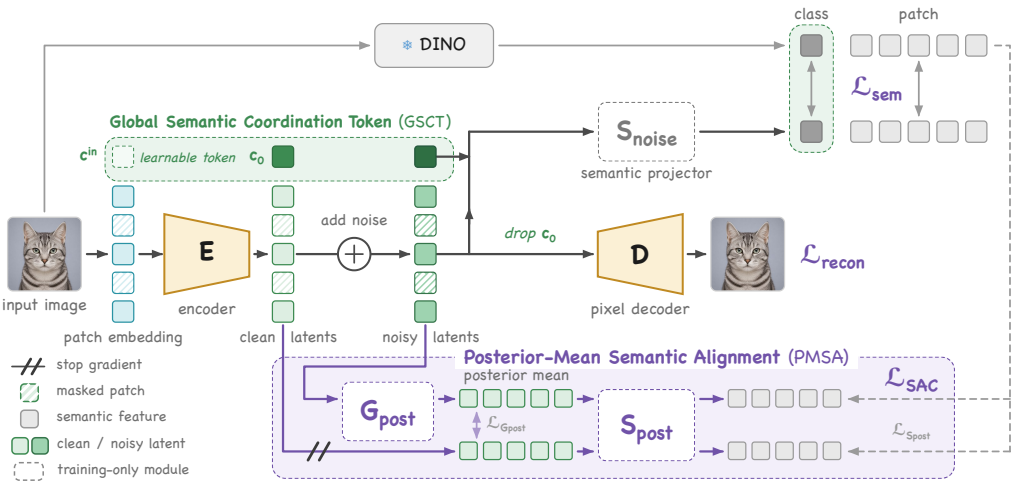

Figure 2: **Overall framework of AffineTok.** The input image is processed in parallel by a frozen VFM and a trainable tokenizer. Its encoder jointly maps patch embeddings and a prepended learnable GSCT to a global output and clean patch latents z_0. After z_0 is noised into z_t, the resulting global and noisy patch representations receive semantic supervision; the global output is then discarded, while z_t is decoded for reconstruction. Along the PMSA path, G_{post} maps z_t to a posterior-mean estimate, and S_{post} maps the estimate to semantics. Only clean patch latents are retained downstream.

- We introduce the tokenizer-side metric $\mathcal{M}_{\text{SAC}}$, a proxy for SAC based on interpolation between neighboring clean latents. By measuring both the coefficient and direction of decoded semantic transitions, it closely tracks generation quality across tokenizers and diffusion model scales without training downstream diffusion models.
- We develop a SAC-guided tokenizer-training method with GSCT and PMSA. GSCT coordinates the organization of clean patch latents across images, keeping cross-image semantic averaging meaningful, while PMSA predicts posterior-mean latents from noisy inputs and supervises their semantics. Both components are used during tokenizer training, preserving the downstream diffusion training unchanged.

2 Method

We study how semantic supervision in visual tokenizers supports semantic recovery required by downstream diffusion. In Sec. 2.1, we decompose the semantic recovery error into the semantic uncertainty in a noisy latent and the gap between the semantics directly predicted from the noisy latent and those decoded from the posterior-mean clean latent. We define their agreement as *Semantic Affine Consistency* (SAC). In Sec. 2.2, we introduce the tokenizer-side metric $\mathcal{M}_{\text{SAC}}$, a proxy for SAC based on interpolation between neighboring clean latents. In Sec. 2.3, we use this analysis to guide tokenizer training by extending semantic supervision to previously unconstrained posterior-mean estimates. The training framework is shown in Fig. 2.

2.1 Semantic Affine Consistency

The goal of denoising is to recover the clean image from a noisy observation. In latent diffusion, this requires recovering not only the clean latent but also the semantic content of the image obtained by decoding it. We refer to the latter objective as *semantic recovery*.

Specifically, given an image $X \sim p_{\text{data}}$, we define

$$z_0 = E(X), \qquad Y_X = \phi(X), \qquad z_t = (1-t)z_0 + t\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{1}$$

where E and D are the tokenizer's encoder and decoder, and ϕ is a frozen visual foundation model.

Under squared denoising loss, the Bayes-optimal clean-latent prediction from z_t is the posterior mean

$$m_{z_0}(z_t, t) := \mathbb{E}[z_0 \mid z_t, t]. \tag{2}$$

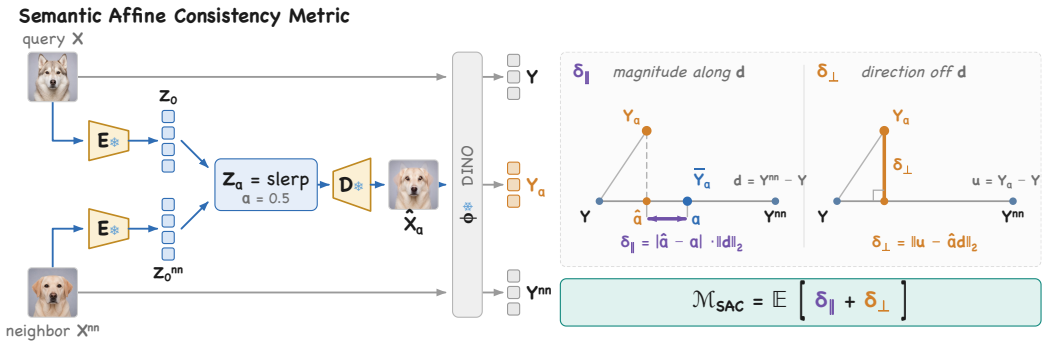

Figure 3: **SAC probing.** We compute $\mathcal{M}_{\text{SAC}}$ using neighboring latent interpolation to measure the agreement between the average of semantics and the semantics decoded from the averaged latent.

Decoding this prediction and applying ϕ gives the corresponding semantic prediction

$$Y_{m_{z_0}}(z_t, t) := \phi \circ D(m_{z_0}(z_t, t)) \,. \tag{3}$$

Therefore, the error of the *semantic recovery* objective at (z_t, t) under squared semantic loss is

$$\mathcal{E}_{\text{sem}}(z_t, t) := \mathbb{E}\Big[\big\| Y_X - Y_{m_{z_0}}(z_t, t) \big\|_2^2 \,\Big|\, z_t, t \Big] \,. \tag{4}$$

To further analyze this error, we first separate the component caused by insufficient semantic information contained in z_t after noising. To do so, we consider the theoretically optimal semantic prediction given (z_t, t). Under squared semantic loss, this prediction is the conditional mean

$$Y_{z_t}(z_t, t) := \arg\min_y \mathbb{E}\big[\|Y_X - y\|_2^2 \,\big|\, z_t, t \big] = \mathbb{E}[Y_X \mid z_t, t]. \tag{5}$$

The semantic recovery error admits the following exact decomposition (proof in Appendix A):

$$\mathcal{E}_{\text{sem}}(z_t, t) = \mathbb{E}\Big[\|Y_X - Y_{z_t}(z_t, t)\|_2^2 \,\Big|\, z_t, t \Big] + \big\| Y_{z_t}(z_t, t) - Y_{m_{z_0}}(z_t, t) \big\|_2^2 \,. \tag{6}$$

Previous methods introduce a projector P to predict the clean-image semantics from noisy latents (Shi et al., 2026b; Page et al., 2026), using:

$$\mathcal{L}_{\text{noise}} = \|Y_X - P(z_t, t)\|_2^2 \,. \tag{7}$$

For a fixed encoder E, the optimal projector satisfies $P^\star(z_t, t) = \mathbb{E}[Y_X \mid z_t, t] = Y_{z_t}(z_t, t)$. Substituting $P^\star$ into Eq. (7) shows that its minimum is exactly the expectation of the first term in Eq. (6). Therefore, the loss encourages the encoder to reduce the first term.

However, the second term in Eq. (6) reveals a missing requirement. In fact, $Y_{z_t}(z_t, t)$ is the *average of clean-image semantics*, whereas $Y_{m_{z_0}}(z_t, t)$ is the *semantic of an averaged clean latent*. As the semantic decoding map $g = \phi \circ D$ is generally nonlinear, semantic decoding and averaging need not commute, so the two need not agree. We call this missing requirement *Semantic Affine Consistency* (SAC). A tokenizer satisfies SAC when $Y_{m_{z_0}}(z_t, t) = Y_{z_t}(z_t, t)$, namely,

$$\phi \circ D(\mathbb{E}[z_0 \mid z_t, t]) = \mathbb{E}[\phi \circ D(z_0) \mid z_t, t]. \tag{8}$$

2.2 SAC Aligns with Downstream Generation Capacity

A natural question is how SAC, a requirement overlooked by existing methods, relates to downstream generation: do tokenizers that better satisfy SAC support stronger downstream generation?

In practice, directly evaluating Eq. (8) would require estimating both $Y_{z_t}(z_t, t)$ and $Y_{m_{z_0}}(z_t, t)$ over the full diffusion posterior. Inspired by (Xu et al., 2026), we approximate local posterior averaging with a two-point support formed by a clean latent z_0 and its nearest neighbor z_0^{nn}. Their interpolated latent z_α serves as a proxy for the posterior mean m_{z_0}. Correspondingly, the semantics decoded from

z_α approximate $Y_{m_{z_0}}$, while the interpolation of their clean-image semantics approximates Y_{z_t}. We therefore probe SAC by comparing these two semantic quantities.

Specifically, let $z_0 = E(X)$ be a query latent and $z_0^{\mathrm{nn}} = E(X^{\mathrm{nn}})$ its nearest neighbor in a latent bank. We fix $\alpha = 0.5$ and form

$$z_\alpha = \mathrm{slerp}\left(z_0, z_0^{\mathrm{nn}}; \alpha\right), \qquad \hat{X}_\alpha = D(z_\alpha). \tag{9}$$

We extract the endpoint semantics and the semantics decoded from the interpolated latent as

$$Y = \phi(X), \qquad Y^{\mathrm{nn}} = \phi(X^{\mathrm{nn}}), \qquad Y_\alpha = \phi(\hat{X}_\alpha). \tag{10}$$

Let d and u denote the semantic differences from Y to Y^{nn} and Y_α, respectively:

$$d = Y^{\mathrm{nn}} - Y, \qquad u = Y_\alpha - Y, \qquad \hat{\alpha} = \frac{\langle u, d\rangle}{\|d\|_2^2}. \tag{11}$$

The target semantic difference is αd. To measure the error in interpolation coefficient and semantic direction, we further decompose the difference between u and αd into components parallel and orthogonal to d:

$$\delta_\parallel = |\hat{\alpha} - \alpha| \cdot \|d\|_2, \qquad \delta_\perp = \|u - \hat{\alpha} d\|_2. \tag{12}$$

As shown in Fig. 3, we define the tokenizer-side SAC metric $\mathcal{M}_{\mathrm{SAC}}$ as

$$\mathcal{M}_{\mathrm{SAC}} = \mathbb{E}_{X, X^{\mathrm{nn}}}\left[\delta_\parallel + \delta_\perp\right]. \tag{13}$$

A lower $\mathcal{M}_{\mathrm{SAC}}$ indicates that latent interpolation more closely preserves the corresponding semantic interpolation. As shown in Fig. 1(b), across 14 visual tokenizers, $\mathcal{M}_{\mathrm{SAC}}$ achieves a Pearson correlation of 0.960 with SiT-XL gFID.

2.3 SAC-Guided Tokenizer Training

Having established $\mathcal{M}_{\mathrm{SAC}}$ as a tokenizer-side probe of SAC, we next ask how SAC can guide tokenizer training. A straightforward approach would be to directly optimize Eq. (13) over sampled interpolation pairs. However, enforcing consistency over this restricted set of latent-space locations does not ensure SAC throughout the latent space.

Before introducing our SAC-guided components, we specify the base objective for tokenizer training:

$$\mathcal{L}_{\mathrm{tok}} = \mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{sem}} \mathcal{L}_{\mathrm{sem}}, \tag{14}$$

where $\mathcal{L}_{\mathrm{rec}}$ aggregates pixel-wise reconstruction loss, LPIPS loss, GAN loss (Van Den Oord et al., 2017), and KL loss (Kingma & Welling, 2013), while $\mathcal{L}_{\mathrm{sem}}$ (Shi et al., 2026b) combines masked feature reconstruction with noisy-latent alignment to frozen VFM features, thereby injecting clean-image semantics into noisy latents and reducing the semantic uncertainty captured by the first term of Eq. (6). While this constrains the direct noisy-latent semantic path $Y_{z_t}(z_t, t)$, it leaves $Y_{m_{z_0}}(z_t, t)$, the semantics recovered through diffusion's clean-latent prediction, outside the training objective.

Completing semantic recovery and satisfying SAC require bringing this missing posterior-mean path under supervision. We therefore introduce two complementary components for tokenizer training. GSCT (Sec. 2.3.1) promotes a consistent organization of clean-latent semantics, keeping cross-image semantic averaging meaningful, while PMSA (Sec. 2.3.2) predicts posterior-mean latents from noisy inputs and explicitly supervises their semantics.

2.3.1 Global Semantic Coordination Token

To make latents more semantic, previous methods align them with VFM features (Yao et al., 2025; Page et al., 2026), while RecTok (Shi et al., 2026b) further combines such alignment with masking to encourage patch latents to capture more global semantic information. While effective, masking does not explicitly coordinate how this global information is incorporated into the patch latents, so semantic organization may become less consistent across images, as shown in Fig. C.2. We therefore introduce a *Global Semantic Coordination Token* (GSCT) to provide such global coordination.

GSCT complements position-wise semantic alignment with a joint constraint on the global information carried by all patch latents. Specifically, let $P = [p^1, \ldots, p^N]$ denote the unmasked patch-token sequence supplied to the encoder. We prepend a learned token c^{in} and obtain

$$[\, c_0; z_0 \,] = E([\, c^{\text{in}}; P \,]), \tag{15}$$

where c_0 is the GSCT output and z_0 contains the patch latents.

A training-only semantic projector reads the full encoder output as $[\, \hat{y}^{\text{gsct}}; \hat{Y}^{\text{patch}} \,] = S_{\text{noise}}([\, c_0; z_0 \,])$. Given the frozen VFM features $\phi(X) = [\, y^{\text{cls}}; Y^{\text{patch}} \,]$, the patch predictions retain their original semantic supervision, while the GSCT prediction is aligned with the global VFM feature, (e.g. DINO's class token):

$$\mathcal{L}_{\text{GSCT}} = d_{\cos}\big(\hat{y}^{\text{gsct}}, \text{sg}[y^{\text{cls}}]\big)\,. \tag{16}$$

Because c_0 exchanges information with all patch tokens during encoding, its alignment with the global VFM feature jointly guides how patch latents capture global information under masking. This shared semantic reference coordinates the semantic organization of clean latents across images, keeping cross-image semantic averaging meaningful. The GSCT output is discarded before pixel decoding, leaving the downstream latent interface unchanged.

2.3.2 Posterior-Mean Semantic Alignment

So far, we have injected semantic information into z_t through $\mathcal{L}_{\text{sem}}$ and further ensured consistent semantic coordinates across images through $\mathcal{L}_{\text{GSCT}}$. However, as shown in Sec. 2.1, we still need to explicitly constrain the semantics decoded from the clean-latent prediction, namely, $Y_{m_{z_0}}(z_t, t)$.

In fact, directly constraining $Y_{m_{z_0}}$ raises two practical issues: (1) how to estimate the posterior mean from a noisy latent; and (2) how to avoid the heavy computation through the pixel decoder and frozen VFM at every training step. To address these issues, we introduce two training-only modules: a posterior predictor G_{post} and a semantic projector S_{post}. The former estimates the posterior mean from a noisy latent, while the latter approximates the decoded VFM semantics directly in latent space. The three PMSA training objectives are summarized in Fig. C.1 (Appendix C.1).

Estimate the posterior mean. Our first target is to estimate the posterior mean $m_{z_0}(z_t, t)$ required by Eq. 8 from the noisy latent z_t. We follow the principle used in diffusion models: *supervision from individual clean samples is sufficient to estimate the posterior mean from a noisy input* (Karras et al., 2022). Specifically, although each z_t is paired with only one clean target z_0, the population optimum of squared regression is exactly the posterior mean. We therefore train G_{post} with

$$\mathcal{L}_{G_{\text{post}}} = \lambda_z \left\| G_{\text{post}}(z_t, t) - z_0 \right\|^2, \qquad G^*_{\text{post}}(z_t, t) = \mathbb{E}[z_0 \mid z_t, t] = m_{z_0}(z_t, t). \tag{17}$$

Calibrating the semantic projector. Our second target is to approximate the decoded VFM semantics $\phi \circ D(Z)$ directly in latent space, avoiding repeated computation through the pixel decoder and frozen VFM. We learn a semantic projector S_{post} for this purpose. Given a latent sequence z, it predicts one global feature and N patch features:

$$\widehat{Y}(z) = S_{\text{post}}(z) = [\, \hat{y}^0; \hat{y}^1, \ldots, \hat{y}^N \,], \qquad Y_X = [\, y^0; y^1, \ldots, y^N \,], \qquad y^0 = y^{\text{cls}}. \tag{18}$$

We measure their semantic distance and calibrate S_{post} on observed clean latents using

$$\mathcal{D}_{\text{sem}}\left(\widehat{Y}, Y_X\right) = d_{\cos}(\hat{y}^0, y^0) + \frac{1}{N}\sum_{p=1}^{N} d_{\cos}(\hat{y}^p, y^p), \qquad \mathcal{L}_{S_{\text{post}}} = \lambda_y \mathcal{D}_{\text{sem}}(S_{\text{post}}(z_0), Y_X)\,. \tag{19}$$

The calibrated semantic projector is then used to approximate $\phi \circ D$ on posterior-mean estimates.

Semantic supervision for posterior-mean estimates. With G_{post} estimating $m_{z_0}(z_t, t)$ and S_{post} approximating its decoded VFM semantics, their composition

$$\widehat{Y}_{m_{z_0}}(z_t, t) := S_{\text{post}}(G_{\text{post}}(z_t, t)) \tag{20}$$

provides a latent-space approximation to $Y_{m_{z_0}}(z_t, t)$.

Once z_t retains sufficient semantic information, as encouraged by $\mathcal{L}_{\text{sem}}$, directly supervising $\widehat{Y}_{m_{z_0}}(z_t, t)$ with the clean semantics Y_{z_0} drives its optimum toward $Y_{z_t}(z_t, t)$. We therefore use Y_{z_0} as the supervision target for simplicity

$$\mathcal{L}_{\text{SAC}} = \lambda_y \mathcal{D}_{\text{sem}}(S_{\text{post}}(G_{\text{post}}(z_t, t)), Y_X). \tag{21}$$

By directly supervising the previously unconstrained $Y_{m_{z_0}}$, this objective promotes SAC and improves semantic recovery. We retain $\mathcal{L}_{\text{sem}}$ throughout training because its direct semantic supervision on z_t is easier to learn, thereby reducing the optimization difficulty of this additional constraint.

Overall PMSA objective. The complete PMSA objective combines the three losses:

$$\mathcal{L}_{\text{PMSA}} = \mathcal{L}_{G_{\text{post}}} + \mathcal{L}_{S_{\text{post}}} + \mathcal{L}_{\text{SAC}}. \tag{22}$$

Within PMSA, the three objectives update different parameter sets:

$$\Delta\theta_G = -\eta_G \nabla_{\theta_{G_{\text{post}}}} \mathcal{L}_{G_{\text{post}}}, \Delta\theta_S = -\eta_S \nabla_{\theta_{S_{\text{post}}}} \mathcal{L}_{S_{\text{post}}}, \Delta\theta_E = -\eta_E \nabla_{\theta_E} \mathcal{L}_{\text{SAC}}, \tag{23}$$

where the remaining parameter sets are held fixed in each update.

2.3.3 Overall Objective

The complete tokenizer training objective is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{tok}} + \lambda_{\text{GSCT}} \mathcal{L}_{\text{GSCT}} + \lambda_{\text{PMSA}} \mathcal{L}_{\text{PMSA}}. \tag{24}$$

3 Experiments

We first describe the experimental setup in Sec. 3.1. Then, we evaluate whether $\mathcal{M}_{\text{SAC}}$ predicts downstream generation performance in Sec. 3.2, examine whether GSCT and PMSA improve tokenizer diffusability in Sec. 3.3, and present controlled ablations in Sec. 3.4.

3.1 Experimental Setup

Metric evaluation. Following Xu et al. (2026), we evaluate 14 publicly available visual tokenizer checkpoints spanning different latent dimensions, architectures, and training objectives. We use the SiT-B and SiT-XL gFID results reported in that work to measure downstream generation quality. All tokenizer-side metrics are computed on the ImageNet validation split. We report Pearson, Spearman, and Kendall correlations between each tokenizer-side metric and gFID.

Tokenizer training. We follow the tokenizer architecture and training protocol of RecTok (Shi et al., 2026b), and use DINOv3-S(Siméoni et al., 2025) as the VFM. With a $16\times$ spatial compression ratio, the tokenizer represents each image using $d = 128$ latent dimension. We set $\lambda_{\text{GSCT}} = \lambda_{\text{PMSA}} = 1$. The architectures of G_{post} and S_{post} are detailed in Appendix C.1. GSCT and PMSA are used only during tokenizer training, and the exported tokenizer retains the same latent interface as RecTok.

Diffusion training and evaluation. For downstream diffusion training, we use the same DiT^{DH}-XL (Zheng et al., 2026) backbone and training protocol as RecTok. All experiments are conducted on ImageNet-1K (Russakovsky et al., 2015) at 256×256 resolution. We report Fréchet Inception Distance (gFID) (Heusel et al., 2017), Inception Score (IS) (Salimans et al., 2016), precision (Prec.), and recall (Rec.) (Kynkäänniemi et al., 2019), computed from 50K generated samples following the RecTok evaluation protocol.

3.2 The SAC Metric Predicts Downstream Generation

The SAC metric consistently predicts tokenizer diffusability. As shown in Table 1, $\mathcal{M}_{\text{SAC}}$ achieves the strongest correlation with downstream gFID for both SiT-B and SiT-XL under every correlation measure, with its Pearson correlation reaching 0.960 for SiT-XL. The strong rank correlations further show that it reliably recovers how different latent spaces rank in downstream generation performance.

Separating parallel and orthogonal errors improves prediction. To isolate the effect of the directional decomposition, we compare $\mathcal{M}_{\text{SAC}}$ with $\mathcal{M}_{\text{SAC}_{\ell_2}}$, a basic variant that directly measures $\|Y_\alpha - Y_\alpha^\star\|_2$. As shown in Table 1, even this basic variant correlates more strongly with downstream gFID than iFID across both model scales. Decomposing the same error into parallel and orthogonal components further distinguishes miscalibrated semantic progress from off-direction deviation, leading to more accurate prediction of downstream generation performance.

Table 1: **Correlation between tokenizer-side metrics and downstream gFID.** We report Pearson, Spearman, and Kendall correlations across 14 tokenizers for SiT-XL and 13 tokenizers for SiT-B.

Tokenizer-side metric	SiT-XL			SiT-B		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
iFID	0.896	0.881	0.758	0.856	0.830	0.692
$\mathcal{M}_{\mathrm{SAC}_{\ell_2}}$	0.933	0.943	0.846	0.896	0.874	0.744
$\mathcal{M}_{\mathrm{SAC}}$	**0.960**	**0.974**	**0.912**	**0.946**	**0.929**	**0.821**

3.3 Improved Tokenizer Diffusability

Table 2: **ImageNet** 256 **generation without guidance.** Baseline results follow RecTok (Shi et al., 2026b); guided results are in Appendix D. Dashes indicate unreported values.

Method	Epochs	Params	gFID↓	IS↑	Prec.↑	Rec.↑
Diffusion-centric systems						
DiT Peebles & Xie (2023)	1400	675M	9.62	121.5	0.67	0.67
MaskDiT Zheng et al. (2023)	1600	675M	5.69	177.9	0.74	0.60
SiT Ma et al. (2024)	1400	675M	8.61	131.7	0.68	0.67
REPA Yu et al. (2024)	80	675M	7.94	121.3	0.69	0.64
	800		5.90	157.8	0.70	0.69
DDT Wang et al. (2026)	80	675M	6.62	135.2	0.69	0.67
	400		6.27	154.7	0.68	0.69
Tokenizer-centric systems						
REPA-E Leng et al. (2025)	80	675M	3.46	159.8	0.77	0.63
	800		1.83	217.3	0.77	0.66
VA-VAE Yao et al. (2025)	80	675M	4.29	–	–	–
	800		2.17	205.6	0.77	0.65
AFM Chen et al. (2025)	800	675M	2.04	206.2	0.76	0.67
SVGTok Shi et al. (2026a)	1400	675M	3.36	181.2	–	–
RAE Zheng et al. (2026)	80	839M	2.16	214.8	0.82	0.59
	800		1.51	242.9	0.79	0.63
RecTok Shi et al. (2026b)	20	839M	4.44	149.0	–	–
	80		2.09	198.6	0.79	0.62
	600		1.34	254.6	0.78	0.65
AffineTok (ours)	20	839M	3.28	181.3	0.80	0.57
	80		1.86	217.6	0.80	0.62
	600		**1.21**	**268.4**	0.78	0.67

We then evaluate whether our improvements to the tokenizer facilitate downstream diffusion learning. Our experiments compare downstream generation quality, test whether GSCT and PMSA achieve their intended effects, and assess the reconstruction and representation quality of the tokenizer itself.

Better Diffusability. Table 2 compares our method with RecTok using the same downstream model and training objective, as well as with other generation systems. Our improvements lower gFID throughout training. The benefit is pronounced at 20 epochs, reducing gFID by 26% from 4.44 to 3.28 while increasing IS from 149.0 to 181.3. The improvement persists with extended training, reaching state-of-the-art gFID of 1.21 without guidance after 600 epochs. With guidance, the same checkpoint achieves gFID 1.10 (Appendix D, Table D.1).

GSCT improves semantic consistency across images. We test whether GSCT makes the semantic organization of clean patch latents more consistent across images. We measure the fraction of class-centroid difference directions whose cross-split angle exceeds a fixed threshold (setup details in Appendix C.2). As shown in the left panel of Figure C.2, GSCT reduces the inconsistency rate by 69% on average across all three thresholds, showing that it coordinates the semantic organization of clean patch latents across images.

PMSA improves semantic recovery from predicted clean latents. We test whether PMSA improves the recovery of clean-image semantics from the clean-latent predictions used in denoising. We measure the top-10 semantic retrieval miss rate under severe noise using a semantic projector trained on clean latents and frozen without adaptation, with probe denoisers trained identically for all tokenizers (setup details in Appendix C.3). As shown in the right panel of Figure C.2, GSCT and PMSA reduce the miss rate by 56% on average across the three noise levels, showing that clean-image semantics are recovered more reliably through the predicted-clean-latent path.

3.4 Ablation Studies

We ablate GSCT and PMSA and examine the design of the global semantic coordination token. All variants use the same downstream model and training protocol and are evaluated after 20 epochs.

Contributions of GSCT and PMSA. Table 3 shows that both GSCT and PMSA improve over RecTok when used alone, while combining them further reduces gFID to 3.28. These results are

Table 3: **Component ablation of GSCT and PMSA.** All variants use the same downstream model and training protocol.

Variant	gFID↓	IS↑	Prec.↑	Rec.↑
RecTok	4.44	149.0	0.79	0.57
+ GSCT	3.67	168.38	0.80	0.56
+ PMSA	3.78	162.45	0.80	0.56
+ GSCT + PMSA	**3.28**	**181.28**	0.80	0.57

Table 4: **Role of the global semantic coordination token.** REG denotes encoder-side register tokens. Neither CLS nor register tokens are provided to the downstream DiT.

Setting	REG target	gFID↓	IS↑	Prec.↑	Rec.↑
GSCT (CLS)	None	**3.67**	**168.38**	0.80	0.56
GSCT + 4 REG	None	3.69	167.75	0.80	0.56
GSCT + 4 REG	DINOv3 REG	3.84	161.84	0.80	0.57
4 REG, without GSCT	DINOv3 REG	4.06	157.55	0.80	0.57

consistent with their complementary roles: GSCT coordinates the semantic organization of clean patch latents across images, whereas PMSA directly supervises semantic recovery from predicted clean latents.

Role of global semantic supervision. Table 4 examines whether the benefit of GSCT comes from its global semantic target or merely from adding tokens. Adding four unsupervised register tokens leaves performance nearly unchanged, while supervising them with DINOv3 register features degrades it. Using the supervised register tokens without GSCT performs worse still. These results show that the global DINOv3 $[\mathrm{CLS}]$ feature provides a shared semantic anchor across images, rather than merely increasing the number of encoder tokens.

4 Related Work

Semantically and structurally organized latent spaces. The reconstruction–generation mismatch has motivated visual tokenizers that shape their latent spaces for downstream generative modeling rather than for reconstruction alone. VA-VAE (Yao et al., 2025) aligns projected spatial latents with corresponding features from a frozen vision foundation model (VFM) and matches their within-image pairwise relations. EQ-VAE (Kouzelis et al., 2025) encourages latent maps to be equivariant to anisotropic scaling and rotation through a transformed-latent reconstruction objective. SSVAE (Liu et al., 2025) induces a few-mode bias in the channel eigenspectrum and a low-frequency bias in the spatio-temporal spectrum. UAE (Fan et al., 2025) decomposes clean latent features into a low-frequency base aligned with frozen semantic-encoder features and high-frequency residual bands optimized for pixel-faithful reconstruction. More recently, RecTok (Shi et al., 2026b) distills VFM semantics into rectified-flow states obtained by interpolating clean latents with Gaussian noise. Send-VAE (Page et al., 2026) uses a nonlinear mapper to align similarly perturbed latents with VFM patch features, aiming to improve semantic disentanglement. These methods regularize either clean tokenizer latents or semantic projectors and mappers applied directly to noisy tokenizer states. However, they do not constrain the semantics of the posterior-mean clean latent. We address this component with GSCT, which coordinates clean-latent semantics across images, and PMSA, which supervises the semantics of posterior-mean latent estimates.

Metrics for latent diffusability. Reconstruction quality does not reliably predict downstream generation quality. iFID (Xu et al., 2026) constructs nearest-neighbor latent interpolants and computes FID between their decoded distribution and real images, measuring local manifold continuity. VIV (Zhong et al., 2026) quantifies the irreducible velocity ambiguity induced by overlapping flow-matching trajectories. However, neither considers the semantic recovery problem. By decomposing its error, we identify SAC as a necessary condition and introduce $\mathcal{M}_{\mathrm{SAC}}$, a tokenizer-side proxy for SAC.

5 Conclusion

In this paper, we revisit a fundamental question: *how should a visual tokenizer organize semantics to make downstream diffusion easier to learn?* Existing methods supervise semantics in clean or noisy latents but not those decoded from diffusion's predicted clean latent. Our exact error decomposition identifies the missing requirement as *Semantic Affine Consistency* (SAC). We further operationalize SAC through $\mathcal{M}_{\mathrm{SAC}}$, a tokenizer-side proxy metric that closely tracks generation quality. Guided by this analysis, GSCT coordinates semantics across images and PMSA supervises semantic recovery from posterior-mean latent predictions, without changing downstream diffusion training. Overall, our tokenizer achieves gFID 1.21 without classifier-free guidance and 1.10 with guidance on ImageNet 256×256. We hope our findings foster further research on visual tokenizers explicitly organized for downstream generative modeling.

AI USE STATEMENT

In this work, we used generative AI tools solely for language polishing and suggesting optimizations to existing code. The authors prepared the manuscript draft, and AI assistance with writing was restricted to correcting grammatical errors and improving phrasing. Code assistance was limited to suggesting optimizations to code that had already been developed. We did not use generative AI tools for any other research tasks, including generating research ideas, developing the theoretical formulation or methodology, designing experiments, or interpreting results. We reviewed all AI-assisted work. Specifically, the authors checked all language revisions for accuracy and preservation of the intended meaning, and reviewed and verified all AI-assisted code modifications. We take responsibility for the final content of this work, including the manuscript, accompanying code, and all results produced with the assistance of generative AI.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Bowei Chen, Sai Bi, Hao Tan, He Zhang, Tianyuan Zhang, Zhengqi Li, Yuanjun Xiong, Jianming Zhang, and Kai Zhang. Aligntok: Aligning visual foundation encoders to tokenizers for diffusion models. *arXiv preprint arXiv:2509.25162*, 2025.

Weichen Fan, Haiwen Diao, Quan Wang, Dahua Lin, and Ziwei Liu. The prism hypothesis: Harmonizing semantic and pixel representations via unified autoencoding. *arXiv preprint arXiv:2512.19693*, 2025.

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.

Greg Heinrich, Mike Ranzinger, Hongxu Yin, Yao Lu, Jan Kautz, Andrew Tao, Bryan Catanzaro, and Pavlo Molchanov. Radiov2. 5: Improved baselines for agglomerative vision foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22487–22497, 2025.

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.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)*, pp. 2733–2743, 2019.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

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

Theodoros Kouzelis, Ioannis Kakogeorgiou, Spyros Gidaris, and Nikos Komodakis. Eq-vae: Equivariance regularized latent space for improved generative image modeling. *arXiv preprint arXiv:2502.09509*, 2025.

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.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18262–18272. IEEE, 2025.

Shizhan Liu, Xinran Deng, Zhuoyi Yang, Jiayan Teng, Xiaotao Gu, and Jie Tang. Delving into latent spectral biasing of video vaes for superior diffusability. *arXiv preprint arXiv:2512.05394*, 2025.

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*, pp. 23–40. Springer, 2024.

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.

John Page, Xuesong Niu, Kai Wu, and Kun Gai. Boosting latent diffusion models via disentangled representation alignment. *arXiv preprint arXiv:2601.05823*, 2026.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

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*, pp. 10684–10695, 2022.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

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.

Heiko H Schütt, Alexander D Kipnis, Jörn Diedrichsen, and Nikolaus Kriegeskorte. Statistical inference on representational geometries. *Elife*, 12:e82566, 2023.

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. In *International Conference on Learning Representations*, volume 2026, pp. 154506–154537, 2026a.

Qingyu Shi, Size Wu, Jinbin Bai, Kaidong Yu, Yujing Wang, Yunhai Tong, Xiangtai Li, and Xuelong Li. Rectok: Reconstruction distillation along rectified flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40685–40695, 2026b.

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.

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.

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

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 40633–40642, 2026.

Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Ming-Ming Cheng, et al. Representation entanglement for generation: Training diffusion transformers is much easier than you think. *Advances in Neural Information Processing Systems*, 38:7714–7743, 2026.

Tongda Xu, Mingwei He, Shady Abu-Hussein, Jose Miguel Hernandez-Lobato, Chunhang Zheng, Kai Zhao, Chao Zhou, Ya-Qin Zhang, and Yan Wang. Making reconstruction fid predictive of diffusion generation fid. *arXiv preprint arXiv:2603.05630*, 2026.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15703–15712. IEEE, 2025.

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.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *International Conference on Learning Representations*, volume 2026, pp. 35791–35820, 2026.

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

Tianxiong Zhong, Xingye Tian, Xuebo Wang, Xin Tao, and Pengfei Wan. Diffusing in the right space: A systematic study of latent diffusability. *arXiv preprint arXiv:2606.03578*, 2026.

APPENDIX

A Semantic Recovery and Semantic Affine Consistency **13**

A.1 Proof of the Semantic Recovery Decomposition 13

B Effect of the Feature Extractor on Metric Correlation **14**

C Implementation and Semantic Probe Details **14**

C.1 PMSA Modules and Training Diagram 14

C.2 Cross-Split Semantic-Axis Probe 14

C.3 Posterior-Mean Semantic Recovery Probe 15

D Generation with Guidance **15**

E Reconstruction and Representation Quality **16**

F Additional Related Work **16**

A SEMANTIC RECOVERY AND SEMANTIC AFFINE CONSISTENCY

A.1 PROOF OF THE SEMANTIC RECOVERY DECOMPOSITION

We use the notation of Sec. 2.1 throughout. Fix the encoder E, decoder D, and semantic extractor ϕ.

Proof. By definition, $Y_{z_t}(z_t, t) = \mathbb{E}[Y_X \mid z_t, t]$. The residual $Y_X - Y_{z_t}(z_t, t)$ has zero conditional mean. It is therefore orthogonal in conditional expectation to the discrepancy $Y_{z_t}(z_t, t) - Y_{m_{z_0}}(z_t, t)$, which is fixed given (z_t, t):

$$\begin{aligned} &\mathbb{E}\Big[\big(Y_X - Y_{z_t}(z_t,t)\big)^\top \big(Y_{z_t}(z_t,t) - Y_{m_{z_0}}(z_t,t)\big) \,\Big|\, z_t, t\Big] \\ &\quad = \big(\mathbb{E}[Y_X \mid z_t, t] - Y_{z_t}(z_t,t)\big)^\top \big(Y_{z_t}(z_t,t) - Y_{m_{z_0}}(z_t,t)\big) = 0. \end{aligned} \tag{A.1}$$

Adding and subtracting $Y_{z_t}(z_t, t)$ inside the squared error and using this identity gives

$$\begin{aligned} \mathcal{E}_{\text{sem}}(z_t, t) &= \mathbb{E}\Big[\big\|\big(Y_X - Y_{z_t}(z_t,t)\big) + \big(Y_{z_t}(z_t,t) - Y_{m_{z_0}}(z_t,t)\big)\big\|_2^2 \,\Big|\, z_t, t\Big] \\ &= \mathbb{E}\Big[\|Y_X - Y_{z_t}(z_t,t)\|_2^2 \,\Big|\, z_t, t\Big] + \big\|Y_{z_t}(z_t,t) - Y_{m_{z_0}}(z_t,t)\big\|_2^2, \end{aligned} \tag{A.2}$$

which is Eq. (6). Averaging over (z_t, t) gives the population decomposition.

According to Eq. (A.1), the two error components in Eq. (6) are orthogonal in conditional expectation. Previous methods supervise semantics predicted directly from noisy latents, targeting the first term without explicitly constraining the second. Reducing the first term alone therefore does not ensure SAC, motivating explicit consideration of the second term in tokenizer training. □

B Effect of the Feature Extractor on Metric Correlation

We examine how the choice of frozen feature extractor affects the correlation between $\mathcal{M}_{\text{SAC}}$ and downstream gFID. We replace only the feature extractor used to compute the metric, keeping the tokenizer checkpoints and downstream gFID values fixed. We use patch features, following the DINO-based evaluation. Table B.1 compares DINOv3-S (Siméoni et al., 2025), DINOv2-S (Oquab et al., 2023), SigLIP 2 (Tschannen et al., 2025), and RADIOv2 (Heinrich et al., 2025).

Table B.1: **Feature-extractor ablation for $\mathcal{M}_{\text{SAC}}$.** We report Pearson, Spearman, and Kendall correlations with downstream gFID across 14 tokenizers for SiT-XL and 13 tokenizers for SiT-B.

VFM	SiT-XL			SiT-B		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
iFID	0.896	0.881	0.758	0.856	0.830	0.692
$\mathcal{M}_{\text{SAC}}$ (DINOv3)	**0.960**	**0.974**	**0.912**	**0.946**	**0.929**	**0.821**
$\mathcal{M}_{\text{SAC}}$ (DINOv2)	0.948	0.960	0.868	0.938	**0.929**	**0.821**
$\mathcal{M}_{\text{SAC}}$ (SigLIP 2)	0.911	0.943	0.846	0.900	0.912	0.795
$\mathcal{M}_{\text{SAC}}$ (RADIOv2)	0.948	0.952	0.846	0.938	0.923	0.795

All four VFM-based variants correlate more strongly with downstream gFID than iFID, with DINOv3-S achieving the strongest overall correlations.

C Implementation and Semantic Probe Details

C.1 PMSA Modules and Training Diagram

The posterior predictor G_{post} is a four-block transformer with hidden dimension 768, projecting the noisy latent sequence from d to 768 dimensions and back to d. The semantic projector S_{post} adopts a two-layer transformer architecture with hidden dimension 384.

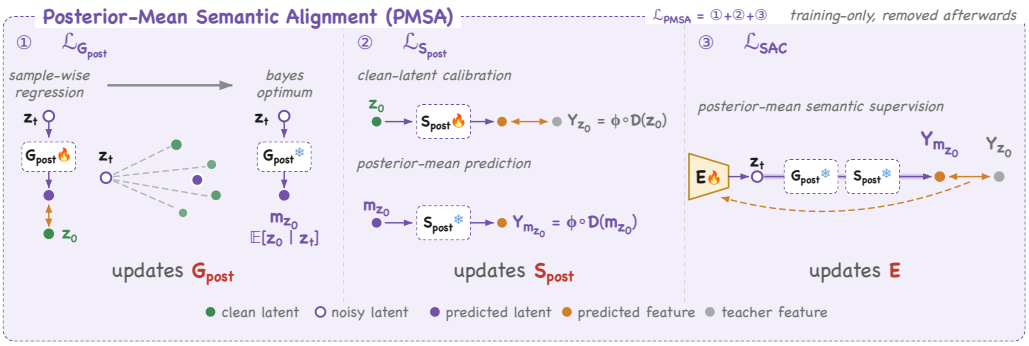

Figure C.1: **PMSA training.** PMSA learns a posterior-mean predictor, calibrates a semantic projector on clean latents, and supervises the semantics recovered from predicted clean latents.

C.2 Cross-Split Semantic-Axis Probe

Following Park et al. (2023), we represent the semantic difference between two classes as the direction between their latent centroids. If the latent semantics are consistent across images, the same class-pair direction estimated from different image subsets should agree. Specifically, we split 20 images from each of 50 ImageNet classes into two halves, compute class-centroid difference directions in each half, and count directions whose cross-split angle exceeds a fixed threshold. To reduce split-induced estimation error, we repeat the analysis over 20 partitions following Schütt et al. (2023).

C.3 Posterior-Mean Semantic Recovery Probe

Following standard probing practice (Alain & Bengio, 2016; Hewitt & Liang, 2019), we train a semantic projector on clean latents and freeze it. We then measure the top-10 semantic retrieval miss rate under severe noise.

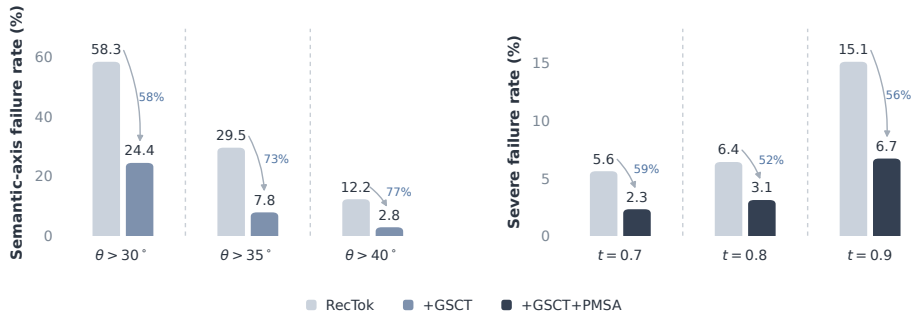

Figure C.2: **Targeted semantic probes for GSCT and PMSA (lower is better).** *Left:* Cross-split semantic-axis inconsistency rates for RecTok and +GSCT across three angular thresholds. *Right:* Top-10 semantic retrieval miss rates for RecTok and +GSCT+PMSA under three severe noise levels.

D Generation with Guidance

Table D.1 complements the unguided results in Table 2. For each method, we report the longest-training checkpoint with available guided results in the comparison. AffineTok is evaluated with a guidance weight of 1.7 at 600 epochs, achieving gFID 1.10.

Method	Epochs	Params	gFID↓	IS↑	Prec.↑	Rec.↑
Diffusion-centric systems						
DiT Peebles & Xie (2023)	1400	675M	2.27	278.2	0.83	0.57
MaskDiT Zheng et al. (2023)	1600	675M	2.28	276.6	0.80	0.61
SiT Ma et al. (2024)	1400	675M	2.06	270.3	0.82	0.59
MDTv2 Gao et al. (2023)	1080	675M	1.58	314.7	0.79	0.65
REPA Yu et al. (2024)	800	675M	1.42	305.7	0.80	0.64
DDT Wang et al. (2026)	400	675M	1.26	310.6	0.79	0.65
Tokenizer-centric systems						
REPA-E Leng et al. (2025)	800	675M	1.26	314.9	0.79	0.66
VA-VAE Yao et al. (2025)	800	675M	1.35	295.3	0.79	0.65
AFM Chen et al. (2025)	800	675M	1.37	293.6	0.79	0.65
SVGTok Shi et al. (2026a)	1400	675M	1.92	264.9	–	–
RAE Zheng et al. (2026)	800	839M	1.13	262.6	0.78	0.67
RecTok Shi et al. (2026b)	600	839M	1.13	289.2	0.79	0.67
AffineTok (ours)	600	839M	**1.10**	282.7	0.77	0.68

Table D.1: **Class-conditional generation on ImageNet** 256 **with guidance.** Epochs and Params denote downstream generator training epochs and parameter count. Prior-method values are taken from RecTok (Shi et al., 2026b). Best gFID is in bold.

E Reconstruction and Representation Quality

Experimental setting. We evaluate tokenizer reconstruction and representation quality on ImageNet-1K at 256×256 resolution. Reconstruction quality is measured by rFID, PSNR, and LPIPS; we also report $\mathcal{M}_{\mathrm{SAC}}$. Following RecTok (Shi et al., 2026b), L.P. Acc. denotes top-1 linear-probing accuracy on frozen latents; *Pooled* averages the patch latents, while *Flat* vectorizes them.

Table E.1: **Tokenizer reconstruction and representation quality on ImageNet-1K.**

Tokenizer	rFID↓	PSNR↑	LPIPS↓	L.P. Acc.↑		$\mathcal{M}_{\mathrm{SAC}}$↓
				Pooled	Flat	
RecTok	**0.48**	**26.16**	**0.1016**	62.32	58.22	52.120
Ours	0.49	25.83	0.1086	**72.63**	**68.13**	**51.495**

Improved latent semantics with comparable reconstruction. Table E.1 reports gains of about 10 percentage points in both pooled and flattened linear-probing accuracy, while reconstruction quality remains comparable. These results show that our method substantially improves latent semantics without materially compromising reconstruction.

F Additional Related Work

Latent diffusion generation. Latent diffusion models (LDMs) reduce the cost of high-resolution synthesis by learning a denoising process in the compact continuous space of a pretrained autoencoder (Rombach et al., 2022). DiT (Peebles & Xie, 2023) replaces convolutional denoisers with scalable transformers over latent patches, while SiT (Ma et al., 2024) places this backbone in a stochastic-interpolant framework that supports diffusion- and flow-based objectives. MaskDiT (Zheng et al., 2023) and MDTv2 (Gao et al., 2023) use masked latent modeling to improve training efficiency and contextual learning, while DDT (Wang et al., 2026) decouples semantic extraction from velocity decoding. Beyond architectural design, REPA (Yu et al., 2024) aligns intermediate denoiser states with clean-image representations from a pretrained visual encoder. REG (Wu et al., 2026) instead entangles low-level image latents with a high-level class token from a pretrained visual encoder during denoising, keeping semantic guidance active at inference. These advances improve the generator after tokenization while leaving the tokenizer-defined latent geometry fixed. Our work instead studies how that geometry should organize semantics for downstream denoising.