Representation Entanglement for Generation: Training Diffusion Transformers Is Much Easier Than You Think

Ge Wu[1,2]   **Shen Zhang**[3]   **Ruijing Shi**[1]   **Shanghua Gao**[4]   **Zhenyuan Chen**[1,2]   **Lei Wang**[1,2]
Zhaowei Chen[3]   **Hongcheng Gao**[5]   **Yao Tang**[3]   **Jian Yang**[1]   **Ming-Ming Cheng**[1,2]   **Xiang Li**[1,2†]

[1]NKIARI, Shenzhen Futian, [2]VCIP, CS, Nankai University, [3]JIIOV Technology,
[4]Harvard University, [5]University of Chinese Academy of Sciences

Figure 1: **Representation Entanglement for Generation demonstrates excellent image quality.**

Abstract

REPA and its variants effectively mitigate training challenges in diffusion models by incorporating external visual representations from pretrained models, through alignment between the noisy hidden projections of denoising networks and foundational clean image representations. We argue that the external alignment, which is absent during the entire denoising inference process, falls short of fully harnessing the potential of discriminative representations. In this work, we propose a straightforward method called *Representation Entanglement for Generation* (**REG**), which entangles low-level image latents with a single high-level class token from pretrained foundation models for denoising. REG acquires the capability to produce coherent image-class pairs directly from pure noise, substantially improving both generation quality and training efficiency. This is accomplished with negligible additional inference overhead, requiring only one single additional token for denoising (<0.5% increase in FLOPs and latency). The inference process concurrently reconstructs both image latents and their corresponding global semantics, where the acquired semantic knowledge actively guides and enhances the image generation process. On ImageNet 256×256, SiT-XL/2 + REG demonstrates remarkable convergence acceleration, achieving **63×** and **23×** faster training than SiT-XL/2 and SiT-XL/2 + REPA, respectively. More impressively, SiT-L/2 + REG trained for merely 400K iterations outperforms SiT-XL/2 + REPA trained for 4M iterations (**10×** longer). Code is available at: `https://github.com/Martinser/REG`.

[†]corresponding author. xiang.li.implus@nankai.edu.cn
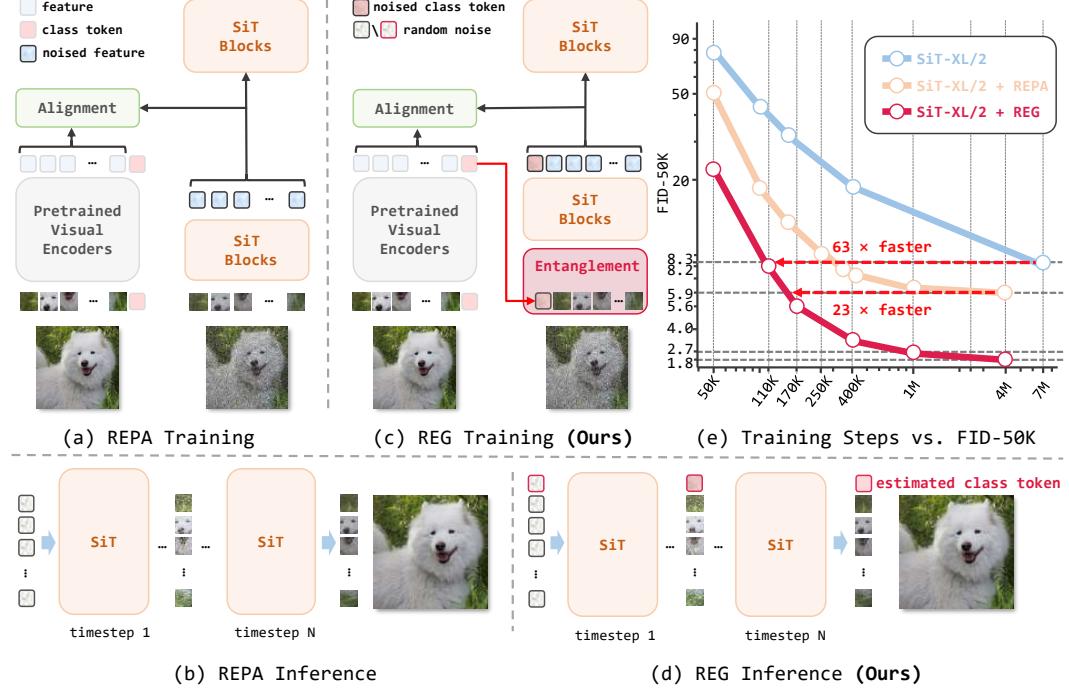
Preprint. Under review.

Figure 2: **Comparison between REPA and our Representation Entanglement for Generation (REG).** (a) During training, REPA [1] indirectly aligns the intermediate denoising features of SiT [2] with pretrained foundation model representations during training. (b) The external alignment of REPA is absent during actual denoising inference, limiting the effectiveness of discriminative information. (c) REG entangles low-level image latents with a pretrained foundation model's class token in training, providing discriminative semantic guidance to SiT. (d) REG's inference process jointly reconstructs both image latents and their associated global semantics from random noise initialization. The incorporated semantic knowledge continues to guide generation, actively enhancing image quality during inference. (e) On ImageNet 256×256, SiT-XL/2 + REG achieves substantial acceleration in convergence, training **63×** and **23×** faster than SiT-XL/2 and SiT-XL/2 + REPA, respectively.

1   Introduction

Generative models have undergone significant evolution [3, 4, 5, 6, 7], demonstrating remarkable success across diverse applications [8, 9, 10, 11, 12, 13, 14]. Recent progress in high-fidelity image synthesis has been driven by several key innovations: Latent Diffusion Models (LDM) [15] introduced a stable two-phase training framework, while Diffusion Transformers (DiT) [16] enhanced scalability through transformer-based architectures. Building upon these developments, Scalable Interpolant Transformers (SiT) [2] further unified the approach through continuous-time stochastic interpolants for diffusion training. Despite these advances, achieving high-fidelity synthesis remains a substantial resource for model convergence. While recent techniques such as masked training [17, 18] and multi-scale optimization [19] partially alleviate computational costs and accelerate model convergence. However, fundamental optimization challenges persist when relying solely on architecture changes.

Recent studies demonstrate that enhanced generative models can acquire more discriminative representations, positioning them as capable representation learners [20, 1, 21]. However, as quantified by CKNNA metrics [22], these features still underperform compared to those from pretrained vision models [23, 24, 25]. This performance gap has motivated approaches leveraging pretrained visual encoder features to accelerate generative model training convergence. For example, REPA [1] employs implicit feature-space alignment between diffusion models and foundation vision models (see Fig. 2(a)), while REPA-E [21] extends this alignment by enabling end-to-end VAE tuning, and quantitatively demonstrates that enhanced alignment (via increased CKNNA scores directly improves generation fidelity.) However, the external alignment of REPA, which is absent during the entire denoising inference process, falls short of fully harnessing the potential of discriminative information

2

(see Fig. 2(b)). We suggest this structure likely impedes further advancements in discriminative semantic learning and overall generative capability.

To address these limitations, we propose a straightforward method called *Representation Entanglement for Generation* (**REG**), an efficient framework that unleashes the potential of discriminative information through explicitly reflows discriminative information into the generation process (see Fig. 2(c)). REG entangles low-level image latents with a single high-level class token from pretrained foundation models during training by applying synchronized noise injection to both of them with spatial concatenation. The denoising inference process concurrently reconstructs both image latents and their corresponding global semantics from random noise initialization, where the acquired semantic knowledge actively guides and enhances the image generation process (see Fig. 2(d)). REG achieves significant improvements in generation quality, training convergence speed, and discriminative semantic learning, all while introducing minimal computational cost through the addition of just one token (less than 0.5% FLOPs and latency in Tab. 4). On class-conditional ImageNet benchmarks at 256×256 resolution (see Fig. 2(e)), SiT-XL/2 + REG achieves 63× and 23× faster training convergence compared to SiT-XL/2 and SiT-XL/2 + REPA, respectively. Notably, SiT-L/2 + REPA trained for 400K iterations surpasses the performance of SiT-XL/2 + REPA trained for 4M iterations (see Tab. 1).

In summary, our specific contributions are as follows:

- We propose **REG**, an efficient framework that entangles low-level image latents with a single high-level class token from pretrained foundation models for denoising.

- REG significantly enhances generation quality, training convergence speed, and discriminative semantic learning while introducing negligible computational overhead.

- On ImageNet generation benchmarks, REG achieves 63× and 23× faster training convergence than SiT and REPA.

2   Related work

Generative models for image generation. Traditional approaches such as DDPM [5] and DDIM [26] perform iterative noise removal in pixel space, while LDM [15] operates in compressed latent spaces through pretrained autoencoders. Architecturally, early U-Net-based diffusion models [5, 27, 15] rely on iterative denoising, whereas modern transformer-based frameworks like DiT [16] and SiT [2] leverage self-attention mechanisms for superior spatial pattern modeling. Despite these advances, existing methods typically require extensive training iterations to achieve convergence. Current acceleration techniques often necessitate significant architectural modifications, such as masked training paradigms [17, 18] or multi-scale optimization strategies [28, 19]. In contrast, we propose REG, which achieves dual improvements in generation quality and training efficiency while introducing minimal inference overhead (requiring just one additional token during denoising). Crucially, REG accomplishes these gains while preserving the original model architecture, demonstrating that superior training dynamics can be achieved without structural compromises.

Generative models as representation learners. Extensive research has established that intermediate features in diffusion models inherently encode rich semantic representations [1, 21], with demonstrated discriminative capabilities across diverse vision tasks including semantic segmentation [29, 30, 31], depth estimation [32], and controllable image editing [33, 34, 35]. Recent advancements have further developed knowledge transfer paradigms from diffusion models to efficient networks through techniques like RepFusion's dynamic timestep optimization [36] and DreamTeacher's cross-model feature distillation [37]. Notably, DDAE [20] confirms that improved diffusion models yield higher-quality representations, establishing a direct correlation between generation capability and representation learning performance. Building upon these insights, we propose to systematically integrate discriminative representations into the generative forward process, enabling persistent discriminative guidance throughout denoising inference.

Generative models with external representations. Prior research [38, 39, 40] has explored augmenting diffusion models through auxiliary components. For example, RCG [41] employs a secondary diffusion model to generate the class token for adaLN-condition [42] in unconditional generation. In contrast, our approach eliminates the need for additional models by leveraging a single class token as part of the input to provide discriminative guidance, simultaneously enhancing both dis-

criminative semantic learning and conditional generation performance. Recent advancements have incorporated visual representations from foundation models to accelerate diffusion training. REPA [1] improves semantic representation quality through feature alignment between early diffusion layers and pretrained vision features, while REPA-E [21] extends this framework by enabling end-to-end VAE tuning. However, these methods rely on external alignment mechanisms that do not take the discriminative representations as the input and denoising, which are unable to produce discriminative representations during inference to guide the generation process. Our proposed REG framework structurally integrates spatial visual representations with semantic class embeddings derived from foundation models. This architectural design enables the denoising phase to concurrently refine localized pattern restoration and holistic conceptual representation, thereby establishing context-aware semantic steering that persists throughout the entire generative process.

3 Method

We propose **REG**, an efficient framework that provides discriminative guidance by entangling image latents with foundation model class token (Fig. 2(c, d)). Section 3.1 covers preliminaries, followed by REG's detailed description in Section 3.2.

3.1 Preliminaries

Our work is based on Scalable Interpolant Transformers (SiT) [2], which provide a unified perspective to understand flow and diffusion models. We first introduce the relevant preliminaries. Flow and diffusion models both leverage stochastic processes to gradually transform Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ into data samples $\mathbf{x}_*$. This process can be unified as

$$\mathbf{x}_t = \alpha_t \mathbf{x}_* + \sigma_t \epsilon, \tag{1}$$

where α_t is a decreasing and σ_t an increasing function of time t. Flow-based models typically interpolate between noise and data over a finite interval, while diffusion models define a forward stochastic differential equation (SDE) that converges to a Gaussian distribution as $t \to \infty$.

Sampling from these models can be achieved via either a reverse-time SDE or a probability flow ordinary differential equation (ODE), both of which yield the same marginal distributions for $\mathbf{x}_t$. The probability flow ODE is:

$$\dot{\mathbf{x}}_t = \mathbf{v}(\mathbf{x}_t, t), \tag{2}$$

where the velocity field $\mathbf{v}(\mathbf{x}, t)$ can be formulated by the conditional expectation:

$$\mathbf{v}(\mathbf{x}, t) = \mathbb{E}[\dot{\mathbf{x}}_t \mid \mathbf{x}_t = \mathbf{x}] = \dot{\alpha}_t \mathbb{E}[\mathbf{x}_* \mid \mathbf{x}_t = \mathbf{x}] + \dot{\sigma}_t \mathbb{E}[\epsilon \mid \mathbf{x}_t = \mathbf{x}]. \tag{3}$$

To synthesize data, we can integrate Eqn. (3) in reverse time, initializing from random Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. This process yields samples from $p_0(\mathbf{x})$, serving as an approximation to the true data distribution $p(\mathbf{x})$. This velocity can be estimated by a model $\mathbf{v}_\theta(\mathbf{x}_t, t)$, which is trained to minimize the following loss function:

$$\mathbb{E}_{\mathbf{x}_*, \epsilon, t}[\|\mathbf{v}_\theta(\mathbf{x}_t, t) - \dot{\alpha}_t \mathbf{x}_* - \dot{\sigma}_t \epsilon\|^2]. \tag{4}$$

The reverse-time SDE can describe the probability distribution $p_t(\mathbf{x})$ of $\mathbf{x}_t$ at time t, which can be expressed as:

$$d\mathbf{x}_t = \mathbf{v}(\mathbf{x}_t, t)dt - \frac{1}{2}w_t \mathbf{s}(\mathbf{x}_t, t)dt + \sqrt{w_t}d\overline{\mathbf{w}}_t, \tag{5}$$

with $\mathbf{s}(\mathbf{x}, t)$ denoting the score that can be computed via the conditional expectation:

$$\mathbf{s}(\mathbf{x}_t, t) = -\sigma_t^{-1}\mathbb{E}[\epsilon \mid \mathbf{x}_t = \mathbf{x}]. \tag{6}$$

The score can be reformulated in terms of the velocity $\mathbf{v}(\mathbf{x}, t)$:

$$\mathbf{s}(\mathbf{x}, t) = \sigma_t^{-1} \cdot \frac{\alpha_t \mathbf{v}(\mathbf{x}, t) - \dot{\alpha}_t \mathbf{x}}{\alpha_t \dot{\sigma}_t - \dot{\alpha}_t \sigma_t}. \tag{7}$$

We can learn the velocity field $\mathbf{v}(\mathbf{x}, t)$ and use it to compute the score $\mathbf{s}(\mathbf{x}, t)$ when using an SDE for sampling.

4

3.2 Representation entanglement for generation

REG training process. Given the clean input image $\mathbf{x}_*$, we obtain image latents $\mathbf{z}_* \in \mathbb{R}^{D_z \times C_z \times C_z}$ via VAE encoder [15] and image feature $\mathbf{f}_* \in \mathbb{R}^{N \times D_{vf}}$ from vision foundation encoder (e.g. DI-NOv2 [23]), where $C_z \times C_z$ denotes the latent spatial resolution, and D_z is the channel dimension. Besides, N represents the number of visual tokens, and D_{vf} is the embedding dimension of vision foundation encoder. In REPA, the absence of the ability to autonomously generate discriminative representations to guide generation in inference may reduce the leverage of discriminative information effectively. We introduce the class token $\mathbf{cls}_* \in \mathbb{R}^{1 \times D_{vf}}$ from the vision foundation model to entangle with image latents for providing the discriminative guidance. Here are the specific details:

We inject noise into both the class token and image latents as a paired input for the SiT forward process [2]. Specifically, given two Gaussian noise samples $\epsilon_z \sim \mathcal{N}(0, \mathbf{I})$ and $\epsilon_{cls} \sim \mathcal{N}(0, \mathbf{I})$ with sizes $\mathbb{R}^{D_z \times C_z \times C_z}$ and $\mathbb{R}^{1 \times D_{vf}}$ respectively, we perform interpolation operations at continuous time $t \in [0, 1]$ as follows:

$$\mathbf{z}_t = \alpha_t \mathbf{z}_* + \sigma_t \epsilon_z; \quad \mathbf{cls}_t = \alpha_t \mathbf{cls}_* + \sigma_t \epsilon_{cls}, \tag{8}$$

This defines intermediate states $\mathbf{z}_t$ (noised latents) and $\mathbf{cls}_t$ (noised class token) in the forward diffusion process, where α_t and σ_t control the generation trajectory. Then, we patchify $\mathbf{z}_t$ into $\mathbf{z}'_t \in \mathbb{R}^{N \times D'_z}$, D'_z is the embedding dimension. Afterwards, the class token $\mathbf{cls}_t$ is projected into the same embedding space via a linear layer to obtain $\mathbf{cls}'_t \in \mathbb{R}^{1 \times D'_z}$. Finally, we concatenate them to form $\mathbf{h}_t = [\mathbf{cls}'_t; \mathbf{z}'_t] \in \mathbb{R}^{(N+1) \times D'_z}$, which serves as the input to the subsequent SiT blocks. We perform alignment at specific transformer layers n, where $n = 4$ for SiT-B/2 + REG and $n = 8$ for all other variants, maintaining consistency with REPA. Specifically, we align the projected hidden state feature $h_\phi(H_t^{[n]}) \in \mathbb{R}^{(N+1) \times D_{vf}}$ with the reference representation $\mathbf{y}_* \in \mathbb{R}^{(N+1) \times D_{vf}}$ which is concatenated by $\mathbf{cls}_*$ and $\mathbf{f}_*$. $\mathbf{h}_t^{[n]} \in \mathbb{R}^{(N+1) \times D'_z}$ denotes output of the n-th SiT block, h_ϕ is a trainable MLP projection, and $\mathrm{sim}(\cdot, \cdot)$ represents the cosine similarity. The alignment loss is defined as:

$$\mathcal{L}_{\mathrm{REPA}}(\theta, \phi) := -\mathbb{E}_{\mathbf{x}_t, \epsilon, t} \left[\frac{1}{N} \sum_{n=1}^{N} \mathrm{sim}(\mathbf{y}_*, h_\phi(\mathbf{h}_t^{[n]})) \right]. \tag{9}$$

In addition to alignment, the training objective includes velocity prediction for both the noised image latents $\mathbf{z}_t$ and class token $\mathbf{cls}_t$. The prediction loss is formulated as:

$$\mathcal{L}_{\mathrm{pred}} = \mathbb{E}_{\mathbf{x}_*, \epsilon, t} \left[\|\mathbf{v}(\mathbf{z}_t, t) - \dot{\alpha}_t \mathbf{z}_* - \dot{\sigma}_t \epsilon_z\|^2 + \beta \|\mathbf{v}(\mathbf{cls}_t, t) - \dot{\alpha}_t \mathbf{cls}_* - \dot{\sigma}_t \epsilon_{cls}\|^2 \right], \tag{10}$$

where $\mathbf{v}(\cdot, t)$ is the velocity prediction function, and $\beta > 0$ controls the relative weighting between the image latents and class token denoising objectives. The final training loss integrates both prediction and alignment objectives, where $\lambda > 0$ governs the relative weight of the alignment loss compared to the denoising loss. Specifically, the total loss $\mathcal{L}_{\mathrm{total}}$ is formulated as:

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{pred}} + \lambda \mathcal{L}_{\mathrm{REPA}}. \tag{11}$$

REG inference process. The framework requires no auxiliary networks to generate the class token. REG jointly reconstructs both image latents and corresponding global semantics from random noise initialization. It acquired semantic knowledge actively guides and enhances the generation quality.

In general, REG demonstrates three key advantages over existing approaches: **(1) Improved utilization of discriminative information.** REG directly integrates discriminative information as part of the input during training, enabling both autonomous generation and consistent application of semantic guidance during inference. This design aims to address a limitation of REPA, which cannot autonomously generate discriminative representations to guide generation in inference. Because it relies on an external alignment mechanism during training to utilize discriminative features, rather than incorporating them as input and applying the corresponding denoising task. **(2) Minimal computational overhead.** This design introduces only a single global class token, providing efficient and effective discriminative guidance while incurring an almost negligible computational overhead of less than 0.5% FLOPs and latency at 256×256 resolution (see Tab. 4). **(3) Enhanced performance across metrics.** REG improves superior performance in generation fidelity, accelerating training convergence, and discriminative semantic learning. As shown in Fig. 2(e), REG achieves up to $23\times$ and $63\times$ faster FID convergence than REPA and SiT, significantly reducing training time. Fig. 3 further shows consistently higher CKNNA scores across training steps, network layers, and timesteps.

5

Table 1: **FID comparison across training iterations for accelerated alignment methods.** All experiments are conducted on ImageNet 256×256 without classifier-free guidance (CFG).

Method	#Params	Iter.	FID↓
SiT-B/2	130M	400K	33.0
+ REPA	130M	400K	24.4
+ REG (ours)	132M	**400K**	**15.2**
SiT-L/2	458M	400K	18.8
+ REPA	458M	400K	9.7
+ REPA	458M	700K	8.4
+ REG (ours)	460M	**100K**	**11.4**
+ REG (ours)	460M	**400K**	**4.6**
SiT-XL/2	675M	400K	17.2
SiT-XL/2	675M	7M	8.3
+ REPA	675M	200K	11.1
+ REPA	675M	400K	7.9
+ REPA	675M	1M	6.4
+ REPA	675M	4M	5.9
+ REG (ours)	677M	**200K**	**5.0**
+ REG (ours)	677M	**400K**	**3.4**
+ REG (ours)	677M	**1M**	**2.7**
+ REG (ours)	677M	**2.4M**	**2.2**
+ REG (ours)	677M	**4M**	**1.8**

Table 2: **Comparison of the performance of different methods on ImageNet 256×256 with CFG.** Performance metrics are annotated with ↑ (higher is better) and ↓ (lower is better).

Method	Epochs	FID↓	sFID↓	IS↑	Pre.↑	Rec.↑
Pixel diffusion						
ADM-U [43]	400	3.94	6.14	186.7	0.82	0.52
VDM++ [44]	560	2.40	-	225.3	-	-
Simple diffusion [45]	800	2.77	-	211.8	-	-
CDM [46]	2160	4.88	-	158.7	-	-
Latent diffusion, U-Net						
LDM-4 [15]	200	3.60	-	247.7	0.87	0.48
Latent diffusion, Transformer + U-Net hybrid						
U-ViT-H/2 [47]	240	2.29	5.68	263.9	0.82	0.57
DiffiT [48]	-	1.73	-	276.5	0.80	0.62
MDTv2-XL/2 [18]	1080	1.58	4.52	314.7	0.79	0.65
Latent diffusion, Transformer						
MaskDiT [49]	1600	2.28	5.67	276.6	0.80	0.61
SD-DiT [50]	480	3.23	-	-	-	-
DiT-XL/2 [16]	1400	2.27	4.60	278.2	**0.83**	0.57
SiT-XL/2 [2]	1400	2.06	4.50	270.3	0.82	0.59
+ REPA	800	1.42	4.70	305.7	0.80	0.65
+ REG (ours)	80	1.86	4.49	**321.4**	0.76	0.63
+ REG (ours)	160	1.59	4.36	304.6	0.77	0.65
+ REG (ours)	480	1.40	**4.24**	296.9	0.77	**0.66**
+ REG (ours)	**800**	**1.36**	4.25	299.4	0.77	**0.66**

4 Experiments

In this section, we investigate three key research questions to evaluate the effectiveness and scalability of REG through comprehensive experimentation:

- **Model performance.** Can REG simultaneously accelerate training convergence and enhance generation quality? (Sec. 4.2)

- **Ablation analysis.** Verify the effectiveness of the REG different designs and hyperparameters. (Sec. 4.3)

- **Discriminative semantics.** Can REG improve the discriminative semantics of generative models? (Sec. 4.4)

4.1 Setup

Implementation details.

We adhere strictly to the standard training protocols of SiT [2] and REPA [1]. Experiments are conducted on the ImageNet dataset [51], with all images preprocessed to 256×256 resolution via center cropping and resizing, following the ADM framework [43]. Each image is encoded into a latent representation $z \in \mathbb{R}^{32 \times 32 \times 4}$ using the Stable Diffusion VAE [15]. Model architectures B/2, L/2, and XL/2 (with 2×2 patch processing) follow the SiT specifications [2]. For comparability, we fix the training batch size to 256 and adopt identical learning rates and Exponential Moving Average (EMA) configurations as REPA [1]. Additional implementation details are provided in the *Appendix*.

Evaluation protocol.

To comprehensively evaluate image generation quality across multiple dimensions, we employ a rigorous set of quantitative metrics including Fréchet Inception Distance (FID) [52] for assessing realism, structural FID (sFID) [53] for evaluating spatial coherence, Inception Score (IS) [54] for measuring class-conditional diversity, precision (Prec.) for quantifying sample fidelity, and recall (Rec.) [55] for evaluating coverage of the target distribution, all computed on a standardized set of 50K generated samples to ensure statistical reliability. We further supplement these assessments with CKNNA [22] for analyzing feature-space characteristics. Sampling follows REPA [1], using the SDE Euler–Maruyama solver with 250 steps. Full evaluation protocol details are provided in the *Appendix*.

Table 3: **Verify the effects of various target representation (Target Repr.) [23, 25], the depth of supervision (Depth), and the loss weight (β).** Experiments employ SiT-B/2 architectures trained for 400K iterations on ImageNet 256×256. Performance metrics (with ↓/↑ denoting preferred directions) are computed using an SDE Euler-Maruyama sampler (NFE=250) without classifier-free guidance. REPA† indicates our local reproduction of the original method's reported results.

Method	Target Repr.	Depth	β	FID↓	sFID↓	IS↑	Pre.↑	Rec.↑
SiT-B/2	-	-	-	33.00	6.46	43.70	0.53	0.63
+ REPA	DINOv2-B	4	-	24.40	6.40	59.90	0.59	0.65
+ REPA†	DINOv2-B	4	-	22.38	6.98	66.65	0.59	0.65
	CLIP-L	4	0.03	21.30	6.51	70.14	0.61	0.63
	DINOv2-B	4	0.03	15.22	6.89	94.64	0.64	0.63
	DINOv2-L	4	0.03	17.36	7.02	89.88	0.63	0.63
	DINOv2-B	2	0.03	18.19	6.67	83.96	0.62	0.64
	DINOv2-B	4	0.03	15.22	6.89	94.64	0.64	0.63
	DINOv2-B	6	0.03	16.31	7.11	91.72	0.63	0.64
+ REG	DINOv2-B	8	0.03	17.31	7.23	87.78	0.63	0.63
	DINOv2-B	4	0.01	15.76	6.69	93.75	0.66	0.61
	DINOv2-B	4	0.02	15.64	6.70	93.86	0.66	0.63
	DINOv2-B	4	0.03	15.22	6.69	94.64	0.64	0.63
	DINOv2-B	4	0.05	16.28	6.97	92.39	0.64	0.64
	DINOv2-B	4	0.10	18.41	7.44	84.79	0.61	0.64

4.2 Improving the performance of generative models

Accelerating training convergence. Tab. 1 provides a detailed comparison between REG, SiT [2], and REPA [1] across multiple model scales on ImageNet 256×256 without CFG. The proposed REG framework consistently achieves the lowest FID scores while substantially accelerating training across all configurations. **For smaller models**, SiT-B/2 + REG outperforms SiT-B/2 + REPA by 9.2 FID points and surpasses SiT-L/2 trained for 400K iterations by 3.6 points. **In the large-scale models**, SiT-L/2 + REG achieves an FID of 4.6 at 400K steps, outperforming both SiT-XL/2 + REPA (4M steps) by 1.3 points and SiT-XL/2 (7M steps) by 3.7 points, while requiring only 10.0% and 5.71% of their respective training costs. Similarly, SiT-XL/2 + REG achieves comparable performance to SiT-XL/2 (7M steps) and REPA-XL/2 (4M steps) in just 110K and 170K steps, respectively, demonstrating 63× and 23× faster convergence (see Fig. 2(e)). At 4M steps, REG achieves a record-low FID of 1.8, demonstrating superior scalability and efficiency across model sizes.

Comparison with SOTA methods. Tab. 2 presents a comprehensive comparison against recent SOTA methods utilizing classifier-free guidance. Our framework achieves competitive performance using the REPA's same guidance interval [56] with significantly reduced training cost. REG matches SiT-XL's quality in just 80 epochs (17× faster than SiT-XL's 1400 epochs) and surpasses REPA's 800-epoch performance at 480 epochs, highlighting its superior training efficiency and convergence properties. Additional experiments in the *Appendix* include the results of more training steps, validating the REG's robustness, scalability, and cross-task generalization.

Computational cost comparison. We compare the computational efficiency of REG and REPA under the same model scale (SiT-XL/2) in Tab. 4. REG introduces only a marginal increase in parameter count (+0.30%) and FLOPs (+0.38%) relative to REPA, while maintaining nearly identical latency (6.21s vs. 6.18s, +0.49%).

Table 4: **Computational cost and performance comparison.** This table compares REPA and REG on ImageNet 256×256, detailing model size, FLOPs, sampling steps, latency, and generation quality metrics. REG achieves substantially better sample quality with negligible increases in computational cost.

Method	#Params	FLOPs↓	Latency (s)↓	FID↓	IS↑
SiT-XL/2 + REPA	675	114.46	6.18	7.90	122.60
SiT-XL/2 + REG	677 (+0.30%)	114.90 (+0.38%)	6.21 (+0.49%)	3.44 (+56.46%)	184.13 (+50.19%)

Despite the minimal computational overhead, REG yields substantial improvements in generation quality, achieving a 56.46% relative reduction in FID, alongside a 50.19% increase in IS. These results demonstrate that REG simultaneously improves generation quality and computational efficiency, highlighting its effectiveness as a general-purpose enhancement for generative models.

4.3 Ablation Studies

Different discriminative guidance. We systematically investigate the impact of different pretrained vision encoders and their corresponding class tokens as target representations in Tab. 3. Among all configurations, DINOv2-B achieves the best performance with the lowest FID (15.22) and highest IS (94.64). Notably, all evaluated target representations consistently surpass the REPA, providing empirical evidence that class tokens derived from self-supervised models enhance generation fidelity.

Alignment depth. As shown in Tab. 3, we compare the effects of applying the REPA loss at different network depths. Our analysis reveals that applying the loss in earlier layers yields superior results, which is consistent with REPA's findings. Notably, our method demonstrates consistent improvements over REPA across all configurations, achieving FID reductions ranging from 4.19 to 7.16 points. We attribute these gains to the direct insertion of the class token, which provides discrete global guidance to all layers. This enables adaptive integration of discriminative semantics throughout the network, in contrast to REPA's indirect supervision mechanism, where only selected features are aligned with the target representation. As a result, REG allows remaining layers to capture richer high-frequency details than REPA, contributing to the observed improvements.

Effect of β. Tab. 3 systematically evaluates the impact of varying the loss weight β, which controls the contribution of the class token alignment loss. Among the tested values, $\beta = 0.03$ achieves the best overall performance across all evaluation metrics. Consequently, this value was adopted as the default parameter for all subsequent experiments.

Entanglement signal variants. Tab. 5 systematically evaluates the impact of different entanglement signals on generation quality through concatenative operation. Concatenating noised latent features with either a learnable token or the average of latent features provides limited improvement, likely due to the lack of rich discriminative semantic information In contrast, incorporating discriminative signals yields substantial gains: averaged DINOv2 features significantly reduce FID to 16.86, while the DINOv2 class token achieves the best performance, lowering FID by 9.18 and increasing IS to 94.64.

Table 5: **Ablation study on different entanglement signals.** All experiments are conducted on ImageNet 256×256, using SiT-B/2 models trained for 400K iterations. This experiment adopts the best configuration from Tab. 3 and focuses solely on the impact of different entanglement signals on generation quality.

Method	FID↓	sFID↓	IS↑
SiT-B/2 + REPA	24.40	6.40	59.90
+ one learnable token	23.31	6.48	63.44
+ avg (latent features)	24.12	6.52	60.78
+ avg (DINOv2 features)	16.86	6.67	84.91
+ DINOv2 class token	**15.22**	**6.69**	**94.64**

These results yield two key insights: (1) high-level discriminative information (class token) substantially enhances generation quality, and (2) the entanglement methodology critically governs performance improvements. The demonstrated efficacy of class token concatenation reveals that global discriminative information effectively regularizes the generative latent space, simultaneously boosting both semantic and output quality while maintaining computational efficiency.

Effectiveness of entanglement alone. Tab. 6 evaluates the impact of incorporating class tokens from various pretrained self-supervised encoders into SiT-B/2 without applying representation alignment. The results demonstrate that class token entanglement alone consistently enhances generation quality, with FID improvements ranging from 0.95 to 6.33 points across all variants. Notably, DINOv2-B delivers optimal performance, achieving a 19.18% FID reduction and 35.86% IS improvement compared to the

Table 6: **Ablation study on different class token entanglement [23, 25] without representation alignment.** This table investigates the effectiveness of class token entanglement in the absence of explicit representation alignment. All experiments are conducted on ImageNet 256×256 at 400K iterations.

Method	Class token	FID↓	sFID↓	IS↑
SiT-B/2	-	33.0	6.46	43.70
	CLIP-L	32.05	6.76	47.61
+ Entanglement	**DINOv2-B**	**26.67**	**6.88**	**59.37**
	DINOv2-L	30.16	6.91	52.86

SiT-B/2 baseline. These findings indicate that the model can effectively leverage high-level semantic guidance from the class token, even in the absence of explicit alignment, highlighting the robustness and general utility of class token-based entanglement for generative modeling.

4.4 Enhancing the discriminative semantic learning of generative models

We systematically measure REG, SiT, and REPA's CKNNA scores across training steps, network layers, and timesteps to assess the discriminative semantics of dense features. For fair comparison, we follow REPA's evaluation protocol: We compute CKNNA scores exclusively between spatially
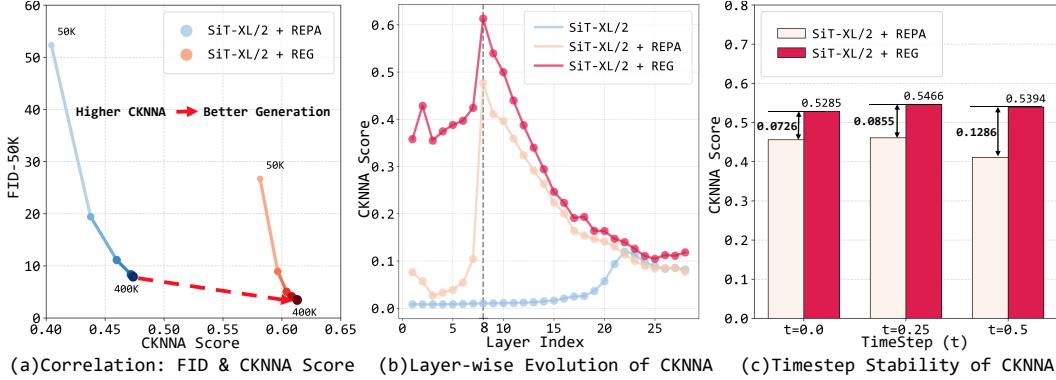
Figure 3: **Analysis of Discriminative Semantics.** (a) Correlation between CKNNA and FID across training steps (color-coded by progression). REG demonstrates superior discriminative semantic learning, achieving higher CKNNA scores alongside lower FID values compared to REPA. (b) Layer-wise CKNNA progression at 400K training steps (t=0.5). REG consistently enhances CKNNA across all network layers, indicating robust discriminative semantics learning. (c) Timestep-wise CKNNA variation at 400K training. REG improves semantic alignment uniformly throughout the training process, outperforming baselines at all timesteps.

averaged generative model dense features and averaged DINOv2-g dense features, while class token are not involved in calculations. Here are the specific situations:

Training steps analysis. Fig. 3(a) shows the positive correlation between CKNNA and FID scores across training steps at layer 8 (t=0.5). It reveals that both REPA and REG achieve improved semantic alignment (higher CKNNA) with better generation quality (lower FID). Notably, REG consistently outperforms REPA in both metrics throughout training, demonstrating its superior capacity for discriminative semantic learning through discriminative semantics guidance.

Layer-wise progression. At 400K training steps (t=0.5) in Fig. 3(b), both REG and REPA exhibit similar CKNNA patterns: semantic scores gradually increase until reaching the peak at layer n=8 (where alignment loss is computed), then progressively decrease. Crucially, REG achieves consistently higher semantic scores than REPA and SiT across all network layers. This improvement stems from REG's innovation of entangling low-level image latents with high-level class token from pretrained foundation models. Through attention mechanisms, REG effectively propagates these discriminative semantics to guide the model in understanding low-level features in early layers, while later layers subsequently focus on predicting high-frequency details.

Timestep robustness. Evaluation of CKNNA at layer 8 (400K steps) demonstrates REG's consistent superiority across all timesteps in Fig. 3(c). This robustness confirms its stable, high-level semantic guidance capability throughout the entire noise spectrum, enabling reliable discriminative semantic performance regardless of noise intensity during generation.

5 Conclusion

This paper presents Representation Entanglement for Generation (REG), a simple and efficient framework that firstly introduces image-class denoising paradigm instead of the current pure image denoising pipeline, which fully unleashes the potential of discriminative gains for generation. REG entangles low-level image latents with a single high-level class token from pretrained foundation models, achieved via synchronized noise injection and spatial concatenation. The denoising process simultaneously reconstructs both image latents and corresponding global semantics, enabling active semantic guidance that enhances generation quality while introducing minimal computational cost through the addition of just one token. Extensive experiments demonstrate REG's superior performance in generation fidelity, accelerating training convergence, and discriminative semantic learning, validating its effectiveness and scalability.

References

[1] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, et al. Representation alignment for generation: Training diffusion transformers is easier than you think. In *arXiv preprint arXiv:2410.06940*, 2024. 2, 3, 4, 6, 7, 14, 15, 16, 17

[2] Nanye Ma, Mark Goldstein, Michael S Albergo, et al. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *ECCV*, 2024. 2, 3, 4, 5, 6, 7, 13, 15, 16

[3] Katherine Crowson, Stella Biderman, Daniel Kornis, et al. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *ECCV*, 2022. 2

[4] Keyu Tian, Yi Jiang, Zehuan Yuan, et al. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *NeurIPS*, 2024. 2

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2, 3

[6] Alexander Tong, Kilian Fatras, Nikolay Malkin, et al. Improving and generalizing flow-based generative models with minibatch optimal transport. In *arXiv preprint arXiv:2302.00482*, 2023. 2

[7] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, et al. Zero-shot text-to-image generation. In *ICML*, 2021. 2

[8] Taihang Hu, Linxuan Li, Joost van de Weijer, et al. Token merging for training-free semantic binding in text-to-image synthesis. In *NeurIPS*, 2024. 2

[9] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, et al. Gem: A generalizable ego-vision multimodal world model. In *arXiv preprint arXiv:2412.11198*, 2024. 2

[10] Hongru Liang, Haozheng Wang, Jun Wang, et al. Jtav: Jointly learning social media content representation by fusing textual, acoustic, and visual features. In *arXiv preprint arXiv:1806.01483*, 2018. 2

[11] Erik Nijkamp, Bo Pang, Hiroaki Hayashi, et al. Codegen: An open large language model for code with multi-turn program synthesis. In *arXiv preprint arXiv:2203.13474*, 2022. 2

[12] Shen Zhang, Yaning Tan, Siyuan Liang, Zhaowei Chen, Linze Li, Ge Wu, Yuhao Chen, Shuheng Li, Zhenyu Zhao, Caihua Chen, et al. Ledit: Your length-extrapolatable diffusion transformer without positional encoding. *arXiv preprint arXiv:2503.04344*, 2025. 2

[13] Taihang Hu, Linxuan Li, Kai Wang, Yaxing Wang, Jian Yang, and Ming-Ming Cheng. Anchor token matching: Implicit structure locking for training-free ar image editing. *arXiv preprint arXiv:2504.10434*, 2025. 2

[14] Tao Liu, Dafeng Zhang, Gengchen Li, Shizhuo Liu, Yongqi Song, Senmao Li, Shiqi Yang, Boqian Li, Kai Wang, and Yaxing Wang. From cradle to cane: A two-pass framework for high-fidelity lifespan face aging. *arXiv preprint arXiv:2506.20977*, 2025. 2

[15] Robin Rombach, Andreas Blattmann, Dominik Lorenz, et al. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 5, 6, 13, 15

[16] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 2, 3, 6, 15, 16

[17] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, et al. Masked diffusion transformer is a strong image synthesizer. In *ICCV*, 2023. 2, 3

[18] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, et al. Mdtv2: Masked diffusion transformer is a strong image synthesizer. In *arXiv preprint arXiv:2303.14389*, 2023. 2, 3, 6, 15

[19] Qihao Liu, Zhanpeng Zeng, Ju He, et al. Alleviating distortion in image generation via multi-resolution diffusion models. In *arXiv preprint arXiv:2406.09416*, 2024. 2, 3

[20] Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *ICCV*. 2, 3

[21] Xingjian Leng, Jaskirat Singh, Yunzhong Hou, et al. Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers. In *arXiv preprint arXiv:2504.10483*, 2025. 2, 3, 4

[22] Minyoung Huh, Brian Cheung, Tongzhou Wang, et al. The platonic representation hypothesis. In *arXiv preprint arXiv:2405.07987*, 2024. 2, 6, 16

[23] Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. Dinov2: Learning robust visual features without supervision. In *arXiv preprint arXiv:2304.07193*, 2023. 2, 5, 7, 8, 13, 15, 16, 17

[24] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 2

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 7, 8

[26] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *arXiv preprint arXiv:2010.02502*, 2020. 3

[27] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 3

[28] Xizhou Zhu, Xue Yang, Zhaokai Wang, et al. Parameter-inverted image pyramid networks. In *arXiv preprint arXiv:2406.04330*, 2024. 3

[29] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, et al. Label-efficient semantic segmentation with diffusion models. In *arXiv preprint arXiv:2112.03126*, 2021. 3

[30] Wenliang Zhao, Yongming Rao, Zuyan Liu, et al. Unleashing text-to-image diffusion models for visual perception. In *ICCV*, 2023. 3

[31] Changyao Tian, Chenxin Tao, Jifeng Dai, et al. Addp: Learning general representations for image recognition and generation with alternating denoising diffusion process. In *arXiv preprint arXiv:2306.05423*, 2023. 3

[32] Ming Gui, Johannes Schusterbauer, Ulrich Prestel, et al. Depthfm: Fast monocular depth estimation with flow matching. In *arXiv preprint arXiv:2403.13788*, 2024. 3

[33] Amir Hertz, Ron Mokady, Jay Tenenbaum, et al. Prompt-to-prompt image editing with cross attention control. In *arXiv preprint arXiv:2208.01626*, 2022. 3

[34] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, et al. Diffusion autoencoders: Toward a meaningful and decodable representation. In *CVPR*, 2022. 3

[35] Zijian Zhang, Zhou Zhao, and Zhijie Lin. Unsupervised representation learning from pre-trained diffusion probabilistic models. In *arXiv preprint arXiv:2211.11036*, 2022. 3

[36] Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *ICCV*, 2023. 3

[37] Daiqing Li, Huan Ling, Amlan Kar, et al. Dreamteacher: Pretraining image backbones with deep generative models. In *ICCV*, 2023. 3

[38] Yongxin Zhu, Bocheng Li, Hang Zhang, et al. Stabilize the latent space for image autoregressive modeling: A unified perspective. In *arXiv preprint arXiv:2410.12490*, 2024. 3

[39] Lijun Yu, Yong Cheng, Zhiruo Wang, et al. Spae: Semantic pyramid autoencoder for multimodal generation with frozen llms. In *NeurIPS*, 2023. 3

[40] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *arXiv preprint arXiv:2501.01423*, 2025. 3

[41] Tianhong Li, Dina Katabi, and Kaiming He. Return of unconditional generation: A self-supervised representation generation method. In *NeurIPS*, 2024. 3

[42] Ethan Perez, Florian Strub, Harm De Vries, et al. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018. 3

[43] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 6, 15, 16

[44] Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. In *NeurIPS*. 6, 15, 16

[45] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. Simple diffusion: End-to-end diffusion for high resolution images. In *ICML*. 6, 15, 16

[46] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *arXiv preprint arXiv:2207.12598*. 6, 15

[47] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *CVPR*. 6, 15

[48] Ali Hatamizadeh, Jiaming Song, Guilin Liu, Jan Kautz, and Arash Vahdat. Diffit: Diffusion vision transformers for image generation. In *ECCV*. 6, 15

[49] Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. In *arXiv preprint arXiv:2306.09305*. 6, 15, 16

[50] Rui Zhu, Yingwei Pan, Yehao Li, Ting Yao, Zhenglong Sun, Tao Mei, and Chang Wen Chen. Sd-dit: Unleashing the power of self-supervised discrimination in diffusion transformer. In *CVPR*. 6, 15

[51] Jia Deng, Wei Dong, Richard Socher, et al. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6

[52] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 6

[53] Charlie Nash, Jacob Menick, Sander Dieleman, et al. Generating images with sparse representations. In *arXiv preprint arXiv:2103.03841*, 2021. 6

[54] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, et al. Improved techniques for training gans. In *NeurIPS*, 2016. 6

[55] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, et al. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019. 6

[56] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *arXiv preprint arXiv:2404.07724*, 2024. 7, 14, 16

[57] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 16

[58] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 16

[59] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, 2019. 16

Representation Entanglement for Generation:
Training Diffusion Transformers Is Much Easier Than You Think
Supplementary Materials

A Discriminative semantics in inference

We posit that REG's semantic reconstruction capability during inference stems from two key design elements: (1) the architectural entanglement of class token with image latents during training, and (2) the consistent application of SiT's [2] velocity prediction loss to both them. Comparative analysis reveals: (1) The One learnable Token (OLT) method (see Fig. 4(a)) concatenates noised latents with a learnable token and only calculates velocity prediction loss $\mathcal{L}_\mathbf{v}$ on dense features. In contrast, REG (see Fig. 4(c)) entangles one high-level noised class token with low-level noised latent features while computing velocity prediction losses $\mathcal{L}_{\text{pred}}$ for both components.
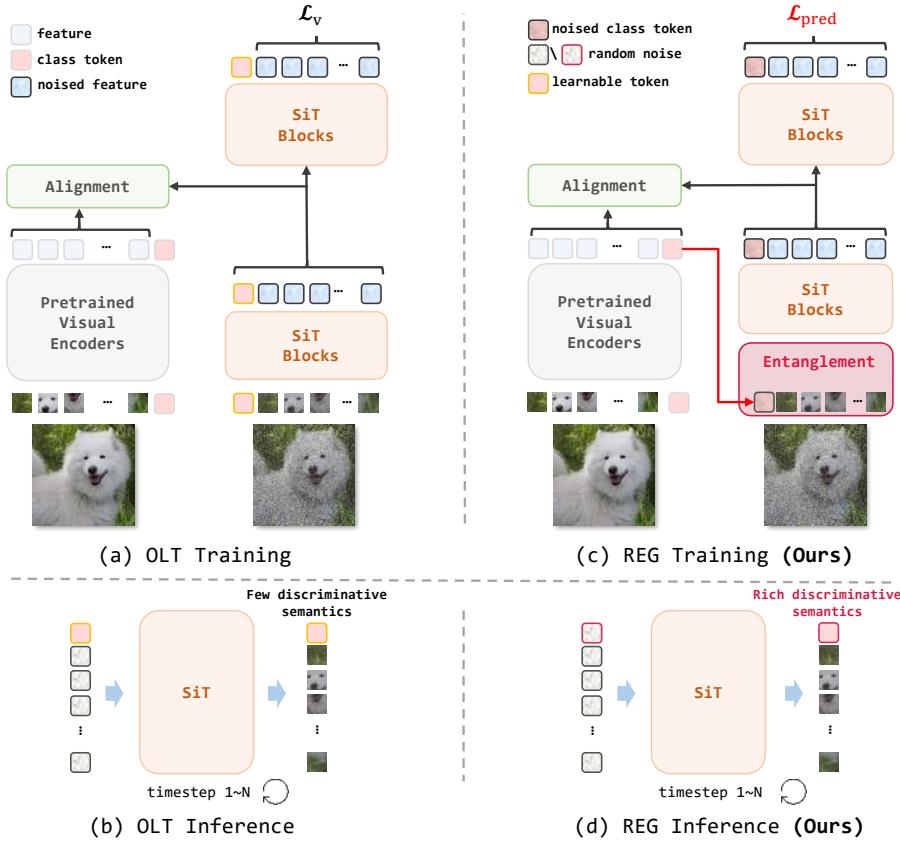

Figure 4: Comparison between the One Learnable Token (OLT) method and REG for generation. (a) During training, OLT simply concatenates noised latents with a learnable token while computing velocity prediction loss $\mathcal{L}_\mathbf{v}$ on dense features. (b) OLT's output learnable token demonstrates minimal discriminative semantics after multi-step denoising. (c) REG entangles one high-level noised class token with low-level noised latent features while computing velocity prediction losses $\mathcal{L}_{\text{pred}}$ for both components. (d) REG can reconstruct the corresponding global semantics of image latents with rich discriminative semantics.

To quantitatively validate the two methods' discriminative semantics in inference, we use 10,000 ImageNet validation images as input processed through identical noise injection via the VAE encoder [15]. REG's inference integrates noised latents with the noise-initialized class token through concatenation before multi-step denoising (see Fig. 4(d)), while OLT similarly processes noised latents with its learnable token (see Fig. 4(b)). Then, we process these ImageNet validation images through DINOv2 [23] to obtain the reference class token. Fig. 5 computes both CKNNA and cosine
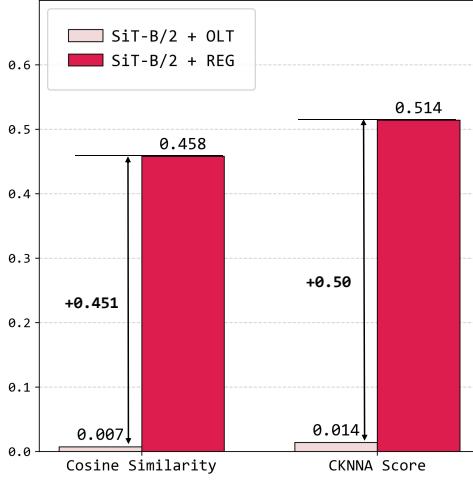
Figure 5: Quantitative evaluation of cosine similarity and CKNNA score between OLT's learnable token, REG's class token, and DINOv2-g's reference token. Results demonstrate REG's superior semantic retention during inference compared to OLT.

similarity between the reference class token and the output from REG's class token/OLT's learnable token using SiT-B/2 as backbone. The results demonstrate a significant disparity: OLT's learnable token achieves only 0.007 cosine similarity and 0.014 CKNNA scores, while REG's class token reaches 0.458 and 0.514, respectively. This empirical evidence confirms REG's superior capacity to preserve discriminative semantics compared to OLT's limited representation capability in inference.

B   Analysis of training overhead in REG

We summarize the total training overhead in Tab. 7, reporting the costs required to reach the same performance upper bounds claimed in the original SiT and REPA papers. All experiments are conducted on 8 NVIDIA A40 GPUs. Our results show that REG requires only 110K training steps to reach the performance level of SiT trained for 7M steps, reducing GPU hours by 98.36%. Moreover, compared with REPA, REG achieves the performance of its 4M counterpart with only 170K iterations, reducing GPU hours by 95.72%. These results highlight the training efficiency of REG, demonstrating faster convergence and significantly lower training overhead compared to prior methods.

Table 7: Training overhead comparison. REG achieves comparable performance to other models while significantly reducing training time.

Model	FID↓	Training step↓	All GPU hours↓
SiT-XL/2	8.3	7M	2380
+ REG (ours)	8.2	110K	39 (-98.36%)
+ REPA	5.9	4M	1800
+ REG (ours)	5.6	170K	77 (-95.72%)

C   256×256 ImageNet

Tab. 8 presents extended training results with CFG using the REPA's same guidance interval [56], demonstrating REG's excellent performance with a 1.40 FID at 480 epochs; it achieves better performance comparable to REPA [1] at 800 epochs while requiring fewer than 40% of the training iterations. In addition, Tab. 9 presents more specific performance details of SiT + REG, further highlighting its superior robustness and accelerated convergence. Tab. 10 presents quantitative performance metrics of SiT-XL + REG under varying classifier-free guidance scale w.

Table 8: Extended REG training on ImageNet 256×256 with CFG demonstrates progressive performance gains.

Method	Epochs	FID↓	sFID↓	IS↑	Pre.↑	Rec.↑
Pixel diffusion						
ADM-U [43]	400	3.94	6.14	186.7	0.82	0.52
VDM++ [44]	560	2.40	-	225.3	-	-
Simple diffusion [45]	800	2.77	-	211.8	-	-
CDM [46]	2160	4.88	-	158.7	-	-
Latent diffusion, U-Net						
LDM-4 [15]	200	3.60	-	247.7	0.87	0.48
Latent diffusion, Transformer + U-Net hybrid						
U-ViT-H/2 [47]	240	2.29	5.68	263.9	0.82	0.57
DiffiT* [48]	-	1.73	-	276.5	0.80	0.62
MDTv2-XL/2* [18]	1080	1.58	4.52	314.7	0.79	0.65
Latent diffusion, Transformer						
MaskDiT [49]	1600	2.28	5.67	276.6	0.80	0.61
SD-DiT [50]	480	3.23	-	-	-	-
DiT-XL/2 [16]	1400	2.27	4.60	278.2	0.83	0.57
SiT-XL/2 [2]	1400	2.06	4.50	270.3	0.82	0.59
+ REPA	800	1.42	4.70	305.7	0.80	0.65
+ REG (ours)	80	1.86	4.49	321.4	0.76	0.63
+ REG (ours)	160	1.59	4.36	304.6	0.77	0.65
+ REG (ours)	300	1.48	4.31	305.8	0.77	0.66
+ REG (ours)	480	1.40	4.24	296.9	0.77	0.66
+ REG (ours)	800	1.36	4.25	299.4	0.77	0.66

Table 9: More performance analysis of SiT + REG across model scales without CFG.

Model	#Params	Iter.	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
SiT-B/2 [2]	130M	400K	33.0	6.46	43.7	0.53	0.63
+ REPA	130M	400K	24.4	6.40	59.9	0.59	0.65
+ REG (ours)	132M	50K	64.7	9.47	23.2	0.40	0.51
+ REG (ours)	132M	100K	36.1	7.74	45.5	0.53	0.61
+ REG (ours)	132M	200K	22.1	7.19	72.2	0.60	0.63
+ REG (ours)	132M	400K	15.2	6.69	94.6	0.64	0.63
SiT-L/2 [2]	458M	400K	18.8	5.29	72.0	0.64	0.64
+ REPA	458M	400K	10.0	5.20	109.2	0.69	0.65
+ REG (ours)	460M	50K	30.1	8.92	52.6	0.58	0.57
+ REG (ours)	460M	100K	11.4	5.36	108.8	0.70	0.60
+ REG (ours)	460M	200K	6.6	5.16	145.4	0.73	0.63
+ REG (ours)	460M	400K	4.6	5.21	167.6	0.75	0.63
SiT-XL/2 [2]	675M	7M	8.3	6.32	131.7	0.68	0.67
+ REPA	675M	4M	5.9	5.73	157.8	0.70	0.69
+ REG (ours)	677M	50K	26.7	16.49	59.2	0.60	0.54
+ REG (ours)	677M	100K	8.9	5.50	125.3	0.72	0.59
+ REG (ours)	677M	200K	5.0	4.88	161.2	0.75	0.62
+ REG (ours)	677M	400K	3.4	4.87	184.1	0.76	0.64
+ REG (ours)	677M	1M	2.7	4.93	201.8	0.76	0.66
+ REG (ours)	677M	2.4M	2.2	4.79	219.1	0.76	0.66
+ REG (ours)	677M	4M	1.8	4.59	230.8	0.77	0.66

D 512×512 ImageNet

To further validate REG's effectiveness, we conduct experiments at 512×512 resolution following REPA's protocol [1]. The RGB images are processed through the VAE [15] to yield 64×64×3 latents, with DINOv2 [23] providing both dense features and class token from 448×448 inputs. As demonstrated in Tab. 11, REG surpasses the performance of REPA trained for 200 epochs and SiT-XL/2 trained for 600 epochs in terms of FID at only 80 epochs, demonstrating its superior effectiveness.

Table 10: The results of SiT-XL + REG at 2.4M training iterations under varying classifier-free guidance scale w, employing the guidance interval method [56].

Model	#Params	Iter.	Interval	w	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
SiT-XL/2 [2]	675M	7M	[0, 1]	1.50	2.06	4.50	270.3	0.82	0.59
+ REG (ours)	675M	2.4M	[0, 0.8]	2.4	1.45	4.32	280.44	0.77	0.67
+ REG (ours)	675M	2.4M	[0, 0.85]	2.4	1.41	4.24	299.65	0.77	0.67
+ REG (ours)	675M	2.4M	[0, 0.9]	2.4	1.61	4.21	334.50	0.79	0.64
+ REG (ours)	675M	2.4M	[0, 0.85]	2.5	1.43	4.25	303.11	0.77	0.67
+ REG (ours)	675M	2.4M	[0, 0.85]	2.4	1.41	4.24	299.65	0.77	0.67
+ REG (ours)	675M	2.4M	[0, 0.85]	2.3	1.40	4.24	296.93	0.77	0.66
+ REG (ours)	675M	2.4M	[0, 0.85]	2.2	1.40	4.25	293.57	0.77	0.67

Table 11: Performance comparison on ImageNet 512×512 with CFG.

Model	Epochs	FID↓	sFID↓	IS↑	Pre.↑	Rec.↑
Pixel diffusion						
VDM++ [44]	-	2.65	-	278.1	-	-
ADM-G, ADM-U [43]	400	2.85	5.86	221.7	0.84	0.53
Simple diffusion (U-Net) [45]	800	4.28	-	171.0	-	-
Simple diffusion (U-ViT, L) [45]	800	4.53	-	205.3	-	-
Latent diffusion, Transformer						
MaskDiT [49]	800	2.50	5.10	256.3	0.83	0.56
DiT-XL/2 [16]	600	3.04	5.02	240.8	0.84	0.54
SiT-XL/2 [2]	600	2.62	4.18	252.2	0.84	0.57
+ REPA	80	2.44	4.21	247.3	0.84	0.56
+ REPA	100	2.32	4.16	255.7	0.84	0.56
+ REPA	200	2.08	4.19	274.6	0.83	0.58
+ REG (ours)	80	**1.68**	**3.87**	**306.9**	0.80	**0.63**

E Experimental setup

Hyperparameter setup. Tab. 12 presents the hyperparameter configurations of SiT + REG across different model scales. Following REPA's experimental protocol [1], we employ AdamW [57, 58] optimization with a batch size of 1×10^{-4} and adopt DINOv2-B [23] as the optimal alignment model, maintaining 250 denoising steps for all inference processes.

CKNNA score. Centered Kernel Nearest-Neighbor Alignment (CKNNA) [22] is a refined metric for measuring representational alignment between neural networks. It extends the traditional Centered Kernel Alignment (CKA) [59] by focusing on local nearest-neighbor kernel alignment, emphasizing the similarity of local topological structures in representation spaces. In the following explanation, we largely adhere to the notation established in the original paper [22]. Mathematically, given two kernel matrices K and L derived from model representations, CKNNA computes alignment through three key steps. First, it applies a truncation function $\alpha(i,j) = 1\left[\phi_j \in \mathrm{knn}\left(\phi_i\right) \wedge \psi_j \in \mathrm{knn}\left(\psi_i\right)\right]$, which selectively preserves pairwise relationships where sample j is among the k-nearest neighbors of i in both models. This creates a sparse kernel matrix that emphasizes local structures. Second, it calculates a weighted covariance of the truncated kernels:

$$\mathrm{Align}_{\mathrm{knn}}(K, L) = \sum_{i,j} \alpha(i,j) \cdot \left(\bar{K}_{ij} \cdot \bar{L}_{ij}\right), \tag{12}$$

where $\bar{K}_{ij}$ and $\bar{L}_{ij}$ denote centered kernel values. Finally, the metric normalizes this alignment score to eliminate scale dependencies:

$$\mathrm{CKNNA}(K, L) = \frac{\mathrm{Align}_{\mathrm{knn}}(K, L)}{\sqrt{\mathrm{Align}_{\mathrm{knn}}(K, K) \cdot \mathrm{Align}_{\mathrm{knn}}(L, L)}}. \tag{13}$$

We adopt REPA's CKNNA computation methodology [1], calculating scores exclusively between spatially averaged dense features from both the generative model and DINOv2-g representations [23]. To ensure fair comparison, the class token is explicitly excluded from all CKNNA calculations.

Table 12: Hyperparameter settings across different model scales.

Backbone	SiT-B	SiT-L	SiT-XL
Architecture			
#Params	132M	460M	677M
Input	32×32×4	32×32×4	32×32×4
Layers	12	24	28
Hidden dim.	768	1,024	1,152
Num. heads	12	16	16
REG settings			
β	0.03	0.03	0.03
λ	0.5	0.5	0.5
Alignment depth	4	8	8
$\text{sim}(\cdot, \cdot)$	cos. sim.	cos. sim.	cos. sim.
Encoder $\mathcal{E}_{VF}(I)$	DINOv2-B	DINOv2-B	DINOv2-B
Optimization			
Batch size	256	256	256
Optimizer	AdamW	AdamW	AdamW
lr	0.0001	0.0001	0.0001
(β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)	(0.9, 0.999)
Interpolants			
α_t	$1 - t$	$1 - t$	$1 - t$
σ_t	t	t	t
w_t	σ_t	σ_t	σ_t
Training objective	v-prediction	v-prediction	v-prediction
Sampler	Euler-Maruyama	Euler-Maruyama	Euler-Maruyama
Sampling steps	250	250	250

F   More visualization results

We present more visualization results of REG in Fig. 6 - 25 with CFG ($w = 4.0$).

Figure 6: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Great white shark" (2).

Figure 7: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Bald eagle" (22).

Figure 8: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Great grey owl" (24).

Figure 9: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Macaw" (88).

Figure 10: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Sulphur-crested cockatoo" (89).

Figure 11: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Koala" (105).

Figure 12: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "American coot" (137).

Figure 13: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Lesser panda" (156).

Figure 14: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Border collie" (232).

Figure 15: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Timber wolf" (269).

Figure 16: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Polecat" (358).

Figure 17: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Lesser panda" (387).

Figure 18: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Giant panda" (388).

Figure 19: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Castle" (483).

Figure 20: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "China cabinet" (495).

Figure 21: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Convertible" (511).

Figure 22: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Bubble" (971).

Figure 23: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Geyser" (974).
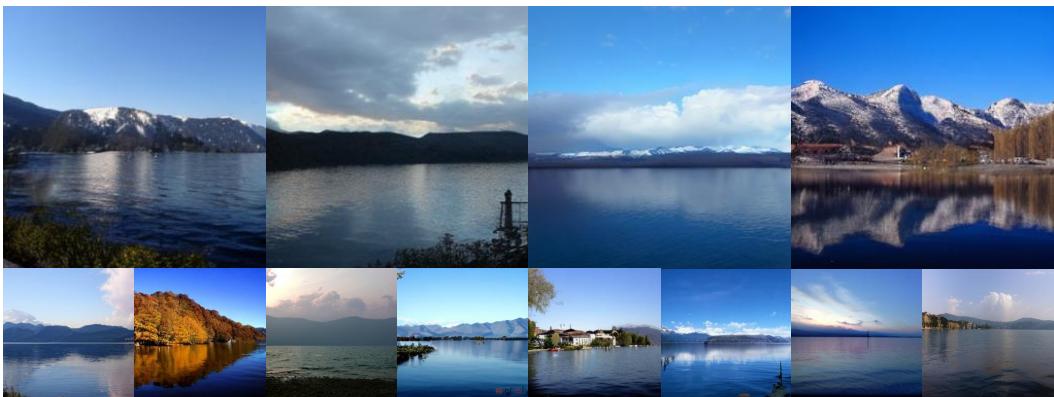

Figure 24: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Lakeside" (975).

Figure 25: The visualization results of SiT-XL/2 + REG use CFG with $w = 4.0$, and the class label is "Volcano" (980).