Self-transcendence: Is External Feature Guidance Indispensable for Accelerating Diffusion Transformer Training?

Lingchen Sun[1,2], Rongyuan Wu[1,2], Zhengqiang Zhang[1,2], Ruibin Li[1], Yujing Sun[1,2], Shuaizheng Liu[1,2], and Lei Zhang[1,2]

[1] The Hong Kong Polytechnic University
[2] OPPO Research Institute

Abstract. Recent works such as REPA have shown that guiding diffusion models with external semantic features (*e.g.*, DINO) can significantly accelerate the training of diffusion transformers (DiTs). However, the use of pretrained external features as guidance signals introduces additional dependencies. We argue that DiTs actually have the power to guide the training of themselves, and propose **Self-Transcendence**, an effective method that achieves fast convergence using internal feature supervision only. The desired internal guidance features should meet two requirements: *structurally clean* to help shallow blocks separate noise from signal, and *semantically discriminative* to help shallow layers learn effective representations. With this consideration, we first align the DiT features with the clean VAE latent features, a native component of latent diffusion, for a short training phase (*e.g.*, 40 epochs) to improve their structural representations, then apply the classifier-free guidance to the intermediate features, enhancing their discriminative capability and semantic expressiveness. These enriched internal features, learned entirely within the model, are used as supervision signals to guide a new DiT training from scratch. Compared to existing self-contained methods, our approach achieves a significant performance boost. It can even surpass REPA, which uses the external DINO features as guidance, in both generation quality and convergence speed for both class-to-image and text-to-image generation tasks. Codes and models can be found at https://github.com/csslc/Self-Transcendence.

Keywords: Diffusion Transformers · Training Acceleration · Internal Guidance · External Guidance

1 Introduction

Diffusion models have emerged as a powerful framework for generative learning, achieving remarkable performance across a wide range of tasks, including image generation [10, 19], video synthesis [25, 30, 42, 49], and multi-modal applications [7, 8, 23, 33]. Despite the great success, training diffusion transformers (DiTs) [26, 32] remains computationally intensive and suffers from slow convergence. Many

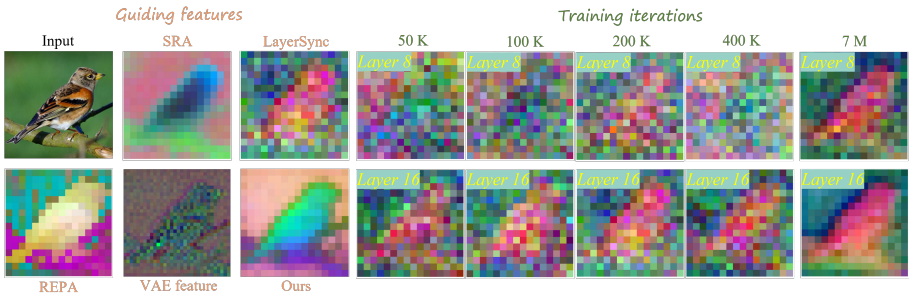

Fig. 1: Left: Comparison of guiding features used in different methods. Our self-contained approach yields more structured and semantically meaningful features than SRA [16] and LayerSync [13], comparable to the external DINO [31] features used in REPA [51]. **Right**: PCA visualization [1] of latent features from both shallow (layer 8) and deeper (layer 16) blocks of SiT with $t = 0.6$ during training. Both layers progressively learn clean and discriminative representations, while the shallow layer learns at a slower pace compared to the deeper one.

methods [13,16,17,21,44,46,47,51,54,55] have been developed to stabilize the DiT model training and accelerate the convergence process. Recent studies [37,44,51] have highlighted the crucial role of meaningful intermediate representations in both improving training efficiency and enhancing generative capability.

To enrich feature representations, several representation learning strategies have been proposed, including masked training [11,12,54,56], contrastive learning [44], and representation alignment [21,43,51]. Among them, the pioneering work REPA [51] introduces an effective regularization strategy to align DiT features with external vision encoders such as DINO [31], significantly accelerating model training and improving generation performance. However, this success is highly dependent on external networks and introduces additional dependencies.

To eliminate the reliance on external supervision, recent works [13,16,44] have explored self-contained alternatives. Dispersive Loss [44] introduces a plug-and-play regularizer that encourages feature dispersion without requiring pre-training or auxiliary data. SRA [16] and LayerSync [13] instead leverage the discriminative features in deeper layers during training to guide the learning of shallower layers. Specifically, SRA employs the EMA model as a teacher during training and performs layer-wise distillation across different noise levels. LayerSync introduces a lightweight synchronization mechanism to align semantically richer and weaker layers. Both of them eliminate the need for external feature extractors. However, their performance lags behind externally-guided approaches such as REPA [51]. Therefore, a critical question arises: *Is external feature guidance indispensable for accelerating diffusion transformer training?*

In this work, we aim to answer this question. We attribute the performance gap to the limited quality of internally generated guidance signals, especially in the early training stages. To serve as effective guidance signals, the features should satisfy two criteria: they should be *structurally clean* to help shallow blocks disentangle noise from signal, and *semantically discriminative* to enable

shallow layers to learn informative representations [37]. As shown on the left side of Fig. 1, when synthesizing an image of a bird, REPA (with DINO features) highlights semantically meaningful regions. By contrast, SRA/LayerSync rely on immature internal features and are semantically less expressive, making them ineffective for guiding shallow layers. Actually, the DiT architecture has the potential to provide useful semantic guidance [20], but this potential has not been fully unleashed. In the right side of Fig. 1, we visualize the latent DiT features of a shallow block (layer 8) and a deeper block (layer 16) using PCA [1] across training. We see that while both layers gradually become more discriminative, the shallow block evolves much more slowly. This indicates that the convergence of DiT is mainly bottlenecked by the learning of clean and semantically rich features in shallow layers, where better guidance signals are important.

Motivated by these observations, we propose **Self-Transcendence**, a fully self-guided training strategy that can surpass REPA-level performance but without external supervision. Self-Transcendence guides DiT training in two stages. The first stage performs **VAE Structure Guidance**. As shown on the left side of Fig. 1, the guiding features of SRA and LayerSync are updated along with training and tend to be noisy and unstable in the early stages. To address this, we directly use the clean VAE latent features within the standard latent diffusion framework to guide the model building structured internal representations. However, while providing a good starting point, the semantic richness of VAE features is limited. Therefore, we perform a second stage of **Self-guided Representation Alignment**. After a warm-up phase, we extract intermediate features from the deeper layer of the partially trained model and apply classifier-free guidance (CFG) to help the guidance feature obtain stronger internal semantics.

This two-stage strategy not only enjoys the benefits of the standard latent diffusion framework (*i.e.*, VAE feature), but is also entirely self-contained, achieving self-transcendence for training DiT models. Training the guiding model adds modest overhead but substantially speeds up subsequent DiT training. Unlike prior methods that rely on external vision encoders [43, 51], our method shows that the internal features of the DiT model itself can serve as an effective guidance. The proposed approach does not require external data and model and is overall more cost-effective. Our contributions are summarized as follows:

- We introduce Self-Transcendence, a fully self-guided training framework to improve DiT model training without relying on external features.
- We propose a two-stage pipeline to obtain a semantically richer internal representation, which first aligns shallow-layer with VAE latent features, and then enhances intermediate features using CFG.
- We show that our method achieves even better performance than REPA in both convergence and generation quality, but without using external data and external features.

2 Related Work

Architectures of Diffusion Models. Early diffusion models typically adopt a U-Net backbone [34], consisting of convolution and attention layers [41]. Re-

cently, Diffusion Transformers (DiTs) [32] replace U-Net with pure transformer blocks, following the design of Vision Transformers [9]. To further enhance DiT, Scalable Interpolant Transformers (SiT) [26] introduce a more flexible interpolant framework to generalize the diffusion process, systematically exploring the design choices of time discretization, prediction targets, interpolant types, and sampling strategies. LightningDiT [50] pushes DiT to its performance limits by incorporating a range of training and architectural optimizations, such as RMSNorm [52], SwiGLU [36], and RoPE [38], enabling faster convergence and more efficient inference. Other efforts investigate architectural improvements such as U-shaped transformer designs [40], dynamic computation adjustment [53], mixture-of-experts [2], linear attention mechanisms [48], and decoupled transformer design [45] to further boost the scalability and efficiency of diffusion models. Our proposed Self-Transcendence presents a new training strategy for the DiT family, guiding the model itself to converge faster.

Acceleration of DiT Training. Recent studies [21, 51] highlight the importance of semantically meaningful representations to improve training efficiency and generation quality. MaskDiT [54] accelerates DiT training by randomly masking 50% of input patches, encouraging efficient learning of informative features. MAETok [4] applies the masking strategy to tokenizer training, improving diffusion models by learning a semantically structured latent space. RCG [24] learns a generative model that generates semantic representations extracted by a self-supervised encoder, using them as conditions for image generation. REPA [51] introduces a representation alignment loss that aligns internal features with pretrained image embeddings [31], significantly boosting training speed and generation performance. The following works explore the use of pretrained vision encoders to provide richer external supervision. U-REPA [39] extends REPA to the U-Net architecture. VA-VAE [50] addresses the optimization bottleneck in latent diffusion by aligning the latent space of a tokenizer with pretrained vision foundation models, enabling faster convergence and better generation quality in high-dimensional settings.

In contrast, some recent works aim to accelerate training without pretrained vision encoders. Dispersive Loss [44] promotes diverse internal representations during diffusion training without external feature guidance. SRA [16] and LayerSync [13] guide shallow layers using semantically richer internal features. However, these methods still lag behind REPA in terms of performance. To bridge this gap, we propose Self-Transcendence, which leverages DiT model's own representations as a substitute for external features.

3 Method

3.1 Motivation and Framework Overview

With the widespread adoption of DiTs [26, 32, 50], accelerating their training has become an important research topic [40, 45, 50, 51, 54]. Generally speaking, shallow blocks of DiT models are responsible for discriminative tasks, *i.e.*, separating clean latent states from noise in the given noisy input, while deeper

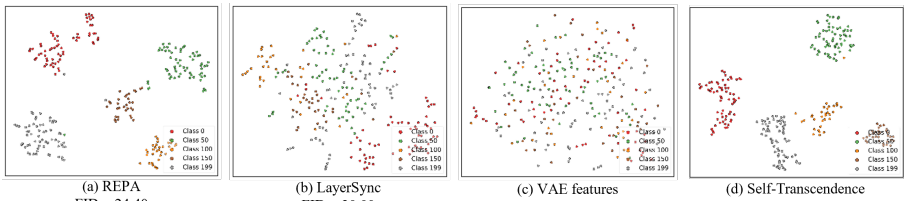

Fig. 2: t-SNE visualizations of the guiding features extracted from (a) REPA [51], (b) LayerSync [13], (c) VAE features, and (d) our Self-Transcendence with $t = 0.4$ in the 200K iteration of SiT-XL/2. Different colors represent different classes. Our internal guiding features demonstrate superior class separability, comparable to REPA.

blocks focus on refining details based on the shallow representations. However, as training progresses, a challenge emerges: shallow layers become slow to learn discriminative features (see Fig. 1) due to the long gradient propagation path. This observation has motivated a line of research [13, 16, 21, 44, 45, 51] to explore how to train shallow blocks with better discriminative representations.

One promising approach is to introduce guiding signals to supervise the learning of shallow features. REPA [51] initiates this research by using external DINO features [31] to guide shallow DiT layers. DINO is a self-supervised vision encoder that learns powerful semantic representations. As shown in Fig. 2(a), DINO embeddings form clear clusters, indicating strong semantic separability. These features boost the learning of shallow DiT layers and significantly accelerate DiT training. However, relying on external DINO features introduces new dependencies: it requires extensive pre-training with external data, which may not always be feasible and desirable. Recent works have begun to explore whether diffusion models can achieve self-acceleration. For example, Layersync and SRA [13, 16] use deeper layer features to guide shallower layer features. However, these features lack stable structural guidance and semantic discriminability, leading to weaker performance, as shown in Fig. 2(b).

In addition to features with discriminative power, we find that features with clean structures [37] in the latent diffusion pipeline, can also be important for guiding features. Although VAE lacks strong discriminative power, its latent space is clean and structured. Surprisingly, VAE features can accelerate DiT training to a level comparable to LayerSync, as shown in Fig. 2(c), suggesting that structured features are effective for providing effective guidance, even without high discriminability.

Based on the above findings, we argue that effective guiding features should meet two criteria: (1) *they should have a clean structure so that they can effectively help shallow layers distinguish noise from signal*, and (2) *they should be semantically discriminative so that they can help shallow layers to learn effective representations.* Several works [6, 20, 22, 27] have pointed out that diffusion models can be leveraged for various discriminative tasks, such as classification. This further motivates us to think whether we can find a discriminative and

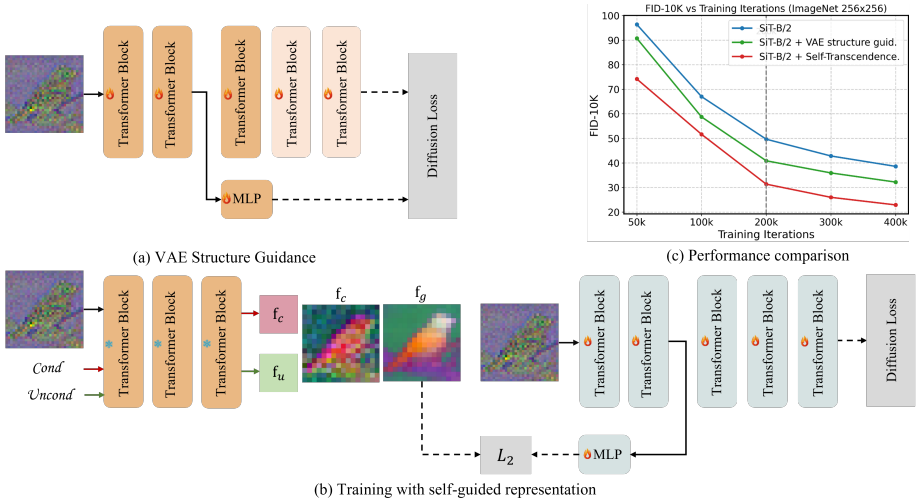

Fig. 3: The framework of our proposed **Self-Transcendence** approach. The spark icon indicates that the parameters of this layer are trainable, while the snowflake icon indicates that they are frozen.

semantically enriched feature representation inside the DiT model to guide the training of itself, achieving self-transcendence.

With these considerations, we propose a new DiT training acceleration framework, namely **Self-Transcendence**, as illustrated in Fig. 3. Unlike one-stage methods such as SRA [16] and LayerSync [13], which update the guiding feature during model training, we adopt a two-stage design, which first warms up a meaningful representation and then uses it to guide a new model learning, for a more stable guiding process. Specifically, in the first stage, we use clean VAE features as guidance to help the model learn a structurally clean representation to distinguish useful information from noise in shallow layers (see Fig. 3(a)). After a certain number of iterations, we freeze this model and use its representation as a fixed teacher. To enhance the semantic expression of the features, we then perform a self-guided representation to better align with the target conditions (see Fig. 3(b)). Compared to REPA that uses external DINO features, our method replaces external features with features from a partially trained internal model, which demonstrates highly competitive guiding performance. Although this warm-up introduces a little extra cost, it significantly accelerates the subsequent model training and improves performance (see Fig. 3(c)). (A one-stage comparison is provided in the **supplementary materials**.)

3.2 VAE Structure Guidance

Standard diffusion models are trained with a single denoising loss imposed on the output, which often leads to slow learning in shallow layers. To improve this, we introduce an auxiliary VAE structure guidance loss to intermediate layers, as shown in Fig. 3(a). Specifically, at the n-th layer, we extract the intermediate

feature $\mathbf{f}_n$, pass it through a lightweight multilayer perceptron (MLP), and align it with the ground-truth VAE latent $\mathbf{z}$ using an L_2 loss, as shown below:

$$\mathcal{L}_{\text{VAE-guide}} = \|\text{MLP}(\mathbf{f}_n) - \mathbf{z}\|_2^2 . \tag{1}$$

The latent representation of the VAE provides a clean structural prior that helps the shallow layers distinguish meaningful signals from noisy inputs.

This alignment can be regarded as an additional diffusion loss constrained only in shallow layers, without adding extra computational resources. Therefore, the shallow blocks are facilitated to perceive structural information and speed up model convergence. As shown in Fig. 3(c), this simple alignment alone can already accelerate the learning of DiT, even obtaining better performance than the existing self-contained methods (please refer to Table 1). However, while VAE structure guidance improves structural learning, it is not as effective as external features (*e.g.*, DINO) for semantic alignment. To address this, we propose a self-guided representation in the following section.

3.3 Self-guided Representation

During the training of a diffusion model, the internal features gradually become more discriminative. Previous works have shown that features from deeper layers often capture stronger semantic information [13, 16, 51]. Therefore, we use deep-layer features as guiding signals to improve the training of a new model, as shown in Fig. 3(b). However, as shown in Figs. 2(a) and (b), even with 200K training steps, the semantic richness of these features still lags behind that of pretrained external vision encoders like DINO.

To address this, we draw inspiration from Classifier-Free Guidance (CFG) [15], a technique widely used in conditional diffusion models. CFG improves the alignment between generated samples and the conditioning input without requiring an external classifier. During training, the model randomly removes the condition to learn both conditional and unconditional denoising. In the inference stage, it combines predictions from both modes using a guidance scale:

$$\mathbf{o}_g = \mathbf{o}_u + \omega \cdot (\mathbf{o}_c - \mathbf{o}_u), \tag{2}$$

where $\mathbf{o}_c$ and $\mathbf{o}_u$ denote diffusion predictions with the condition and uncondition inputs. Increasing ω strengthens the influence of the conditional signal, generating images that better match the desired condition.

Building upon this idea, we extend the CFG from the output space to the feature space. We extract both conditional and unconditional features from the same layer under the same input. Then the two features are combined using a guidance scale, as shown in Eq. (3):

$$\mathbf{f}_g = \mathbf{f}_u + \omega \cdot (\mathbf{f}_c - \mathbf{f}_u), \tag{3}$$

where $\mathbf{f}_c$ and $\mathbf{f}_u$ are the conditional and unconditional features, respectively. Eq. (3) encourages internal representations to align more closely with the desired semantics. As illustrated in Fig. 3(b) and Fig. 2(d), this feature-level guidance

highlights the semantic region and significantly improves the discriminative separability of deep features $\mathbf{f}_g$ compared to their original counterparts $\mathbf{f}_c$.

We then use the enriched feature $\mathbf{f}_g$ to guide the shallower layers together with the standard diffusion loss, *i.e.*, $\mathcal{L} = \mathcal{L}_{diff} + \lambda_{guide} \times \mathcal{L}_{guide}$, as shown in Fig. 3(b). Specifically, intermediate features f_m of the trained DiT are passed through a lightweight MLP (whose architecture is detailed in the **supplementary materials**), and an L_2 loss is applied to align them, as shown below:

$$\mathcal{L}_{guide} = \|\text{MLP}(\mathbf{f}_m) - \mathbf{f}_g\|_2^2 . \tag{4}$$

The self-guided representation explores the intrinsic discriminative ability of diffusion models by applying CFG on deep features to improve the discriminative power of the shallow layers in DiT, further accelerating DiT training, as illustrated in Fig. 3(c).

4 Experiments

4.1 Experimental Setup

Implementation Details. We conduct experiments using two baseline models: SiT [26] with a patch size of 2, and LightningDiT [50] with a patch size of 1. To ensure a fair comparison, we follow the training and inference settings of each baseline. For the proposed Self-Transcendence method, several components should be determined: the guiding model, the guidance scale ω in the self-guided representation, loss weight λ_{guide}, as well as the guided and guiding layers. For all baselines, we use the model trained with the VAE structure guidance at 40 epochs as the guiding model. Note that the guiding model shares the same architecture as the baseline model. We set the guidance scale to $\omega = 30.0$ and $\omega = 10.0$ for the SiT and LightningDiT backbones, respectively. Suppose that the total number of Transformer blocks is n, we select the guided layer as $n/2$ and the guiding layer as $2n/3$. The ablation studies on these parameters are given in Sec. 4.3.

We apply an early stop strategy during training. The self-guided representation loss is only used during the early iterations (20 epochs for base models, 10 epochs for larger models from our experimental study). Then, we remove the self-guided loss and only optimize the diffusion loss. This design aims to improve the training of shallow layers, which lack semantic structures. However, over-training the shallow layers can make the training of deeper layers unstable. Similar phenomena [46] have been observed in REPA. Further discussion is provided in the **supplementary materials**.

Evaluation Metrics. We employ commonly used evaluation metrics [26, 51], including the Fréchet Inception Distance (FID) [14], sFID [29], inception score (IS) [35], precision, and recall [18]. Metrics are computed on 50k samples for Tables 1 and 3, and on 10k samples for Tables 4 and 5.

Compared Methods. We compare with different acceleration methods: REPA [51], Disperse Loss [44], SRA [16], and LayerSync [13]. Various latent diffusion models are also compared. (1) U-Net backbone: LDM [34]. (2) Hybrid

Transformer and U-Net backbone: U-ViT-H/2 [3] and MDTv2-XL/2 [12]. (3) Transformer backbone: MaskDiT [54], SD-DiT [56], DiT-XL/2 [32], and SiT-XL/2 [26].

4.2 Main Results

Comparison with Existing Acceleration Methods. Table 1 shows the comparison between our proposed VAE structure guidance and Self-Transcendence methods with other acceleration approaches. The following observations can be made. **(1)** Firstly, VAE structure guidance alone achieves competitive results with existing self-contained methods such as Disperse Loss and LayerSync. This is because semantic structure is mainly learned during the early training stages. However, Disperse Loss and LayerSync fail to provide strong and stable guidance in this phase. In contrast, our guiding features are derived from a fixed VAE, making it more stable and easier to guide the learning process. **(2)** Secondly, our proposed Self-Transcendence method significantly outperforms all other self-contained techniques. On SiT-XL/2, our method achieves 7.51 FID with only 80 epochs, outperforming the LayerSync trained for 200 epochs (8.80 FID). In addition, Self-Transcendence achieves results comparable to or even better than REPA, which uses external DINO features. **(3)** Thirdly, our approach also brings substantial improvements to LightningDiT, showing that it can be generalized to different backbones (SiT and DiT) and different VAE latent spaces (SD-VAE [34] and VAVAE [50]). Notably, on LightningDiT-XL/1, Self-Transcendence achieves an FID of 3.55 with only 64 epochs of training.

Table 1: Comparisons with different acceleration methods based on the vanilla SiTs and LightningDiTs on ImageNet 256×256. CFG is not used. ↓ denotes that lower values are better.

Model	#Params	Epochs	FID↓
SiT-B/2	130M	120	31.45
SiT-B/2	130M	80	36.14
+ REPA	130M	80	24.40
+ Disperse Loss	130M	80	32.45
+ LayerSync	130M	80	30.00
+ VAE structure guidance (Ours)	130M	80	29.04
+ Self-Transcendence (Ours)	130M	80	**20.49**
SiT-L/2	458M	80	21.41
+ REPA	458M	80	9.70
+ Disperse Loss	458M	80	16.68
+ LayerSync	458M	80	14.83
+ VAE structure guidance (Ours)	458M	80	14.61
+ Self-Transcendence (Ours)	458M	80	**8.74**
SiT-XL/2	675M	800	8.30
SiT-XL/2	675M	120	14.74
SiT-XL/2	675M	80	17.63
+ REPA	675M	80	7.90
+ Disperse Loss	675M	200	10.64
+ LayerSync	675M	200	8.80
+ VAE structure guidance (Ours)	675M	80	12.25
+ Self-Transcendence (Ours)	675M	80	**7.51**
LightningDiT-B/1 (w. VAVAE)	130M	64	15.94
+ VAE structure guidance (Ours)	130M	64	15.87
+ Self-Transcendence (Ours)	130M	64	**14.03**
LightningDiT-XL/1 (w. VAVAE)	675M	64	5.30
+ REPA	675M	64	4.09
+ VAE structure guidance (Ours)	675M	64	5.07
+ Self-Transcendence (Ours)	675M	64	**3.55**

Finally, although our method requires to train a guiding model using VAE structure guidance, this overhead is minimal and worthwhile. As shown in SiT-B/2 and SiT-XL/2, it outperforms the longer-trained baselines by a large margin. This demonstrates that a small cost in warm-up training can lead to significant acceleration and performance gains. Moreover, compared to the widely used

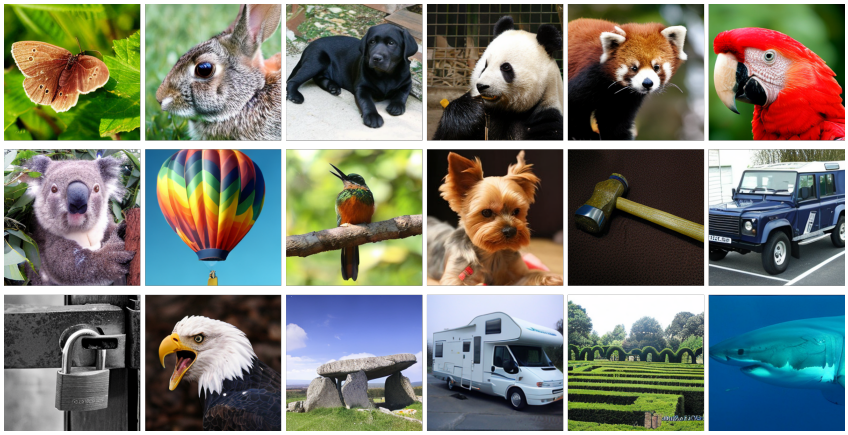

Fig. 4: Examples of generated images on ImageNet 256 × 256 of our proposed Self-Transcendence method. We use classifier-free guidance with scale 4.0.

guiding model (DINO), the training for our guiding model is much easier and more efficient without using external data.

Scalability. We evaluate the proposed Self-Transcendence across different model sizes. As shown in the Table 1, our method consistently accelerates training at all scales. Notably, the performance gain becomes larger as the model size increases. For example, on SiT/B-2, the FID improves from 36.14 (baseline) to 20.49 with a 43.3% relative reduction. On SiT/XL-2, it improves from 17.63 to 7.51, with a 57.4% reduction. This clearly demonstrates the scalability of our method. In addition, our guiding model shares the same backbone as the generation model, which may benefit from larger generation models. We provide qualitative results of SiT-XL/2 on ImageNet at resolutions of 256 using the proposed Self-Transcendence method in Fig. 4, showcasing its ability to generate realistic structures and textures across diverse semantic categories. For animals, such as dog, panda, and bird, our method captures fine-grained details such as fur texture, feather patterns, and facial features with high fidelity. In natural scenes, such as the stone, our trained models generate depth and organic shapes with coherence and realism. For man-made and structured objects such as cars, the generated images exhibit accurate geometry and sharp edges. To further evaluate the scalability of our method, we conduct experiments on ImageNet at a **resolution of 512 × 512**. Experimental details can be found in **supplementary materials**.

We also conduct text-to-image (T2I) generation experiments using the same setting (including training dataset evaluation protocol) as REPA [51]. The results are shown in Table 2. We can see that Self-Transcendence also works well on the more complex T2I generation task, showing better generation scores than REPA (FID: 4.56 vs. 4.90; IS: 33.08 vs. 32.55). Some visual comparisons of the generated images are shown in Fig. 5. We can see that REPA improves over original MMDiT with cleaner structure and fewer artifacts. Our method

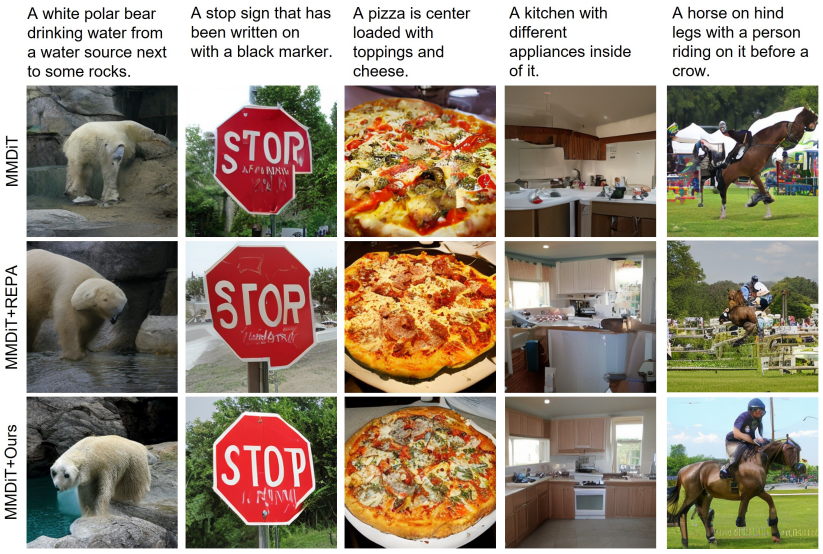

Fig. 5: Qualitative comparison on text-to-image generation (MS-COCO).

further improves prompt alignment and visual performance, yielding more realistic results with faithful structure and illumination.

Table 2: Comparisons of the proposed self-transcendence with REPA in the Text-to-Image task. ↓ and ↑ indicate the lower and higher values are better, respectively.

Model	iters	FID↓	IS↑
MMDiT	150k	6.23	30.74
+ REPA	150k	4.90	32.55
+ Self-Transcendence	150k	**4.56**	**33.08**

Visualization of Training Process. We compare the vanilla SiT model, REPA-enhanced model, and our trained model across 100K to 400K iterations on two ImageNet classes, as illustrated in Fig. 6. Both our model and REPA show faster convergence than the vanilla SiT model, and our method tends to obtain better structural generation. For example, in the shoe class, our method generates more realistic shapes and consistent textures earlier in training. By 400K iterations, our model yields sharper contours and finer details, indicating improved sample quality and stability during training. This can be owed to the fact that the guiding model and the trained model in our method share a similar architecture and learning process, so that the shallow layers are easier to learn the semantic and structural information.

Comparison of Diffusion Models using CFG. We quantitatively compare Self-Transcendence against recent latent diffusion models using two backbones, SiT-XL/2 and LightningDiT-XL, in the Table 3. For the SiT-XL/2 backbone, our method shows significant performance improvement compared with the other self-contained methods (SRA and LayerSync), and achieves similar performance to REPA at 800 epochs in most metrics using just 400 epochs. Specifically, the

Fig. 6: Visual comparison of generated samples from SiT-XL/2 models at different training iterations. For all models, we apply the same seed, noise, and sampling strategy with a CFG scale of 4.0.

baseline SiT-XL achieves a FID of 2.05 and IS of 270.3 after 1400 epochs. Incorporating REPA [51] significantly improves the generation performance, reducing FID to 1.42 and increasing IS to 305.7, with 800 epochs. Our Self-Transcendence framework shows stronger generation acceleration than REPA, achieving an FID of 1.44, IS of 311.3 in only 400 epochs. In addition, leveraging the semantically rich latent space of VAVAE [50], our method outperforms all the compared diffusion models using only 400 epochs in the LightningDiT-XL backbone, with a state-of-art FID result of 1.25.

Table 3: Comparisons across diffusion backbones and acceleration methods on ImageNet 256×256 using CFG. ↓ and ↑ indicate whether lower or higher values are better, respectively.

Model	Epochs	FID↓	sFID↓	IS↑	Pre.↑	Rec.↑
U-Net						
LDM-4	200	3.60	-	247.7	0.87	0.48
Transformer + U-Net hybrid						
U-ViT-H/2	240	2.29	5.68	263.9	0.82	0.57
MDTv2-XL/2	1080	1.58	4.52	**314.7**	0.79	0.65
Transformer						
MaskDiT	1600	2.28	5.67	276.6	0.80	0.61
SD-DiT	480	3.23	-	-	-	-
DiT-XL/2	1400	2.27	4.60	278.2	**0.83**	0.57
SiT-XL/2	1400	2.05	4.50	270.3	0.82	0.59
LightningDiT-XL/1	800	1.35	4.15	295.3	0.79	0.65
Training Acceleration						
SiT-XL/2						
+ REPA	800	1.42	4.70	305.7	0.80	0.65
+ Disperse Loss	≥1200	1.97	-	-	-	-
+ SRA	800	1.58	4.65	311.4	0.80	0.63
+ LayerSync	800	1.89	-	265.3	0.81	0.60
+ Ours	**400**	1.44	4.85	311.3	0.79	**0.66**
LightningDiT-XL/1						
+ Ours	**400**	**1.25**	**4.11**	303.9	0.78	**0.66**

4.3 Ablation Study

We conduct a series of ablation studies to investigate the effectiveness of each component of Self-Transcendence and the selection of hyperparameters. Note that the metrics in the study are calculated on 10,000 samples. **Effectiveness of Each Component**. We ablate Self-Transcendence by removing each key component: VAE structure guidance and self-guided representation. Results are in Table 4. In

variant V1, the model is trained only with the VAE structure guidance loss and the standard diffusion loss. Since VAE features are weak in semantics, removing self-guidance slows convergence and hurts performance. In V2, we use the model trained solely with diffusion loss as the guiding model in the self-guided stage. Without VAE feature guidance, the teacher fails to learn structurally meaningful signals within 40 epochs. Overall, the full model equipped with both components achieves the best performance, confirming the complementary roles of structure guidance and self-guided representation.

Table 4: Ablation studies. All SiT/B-2 models are trained with 80 epochs.

Methods	VAE structure guid.	Self-guided rep.	FID↓	IS↑
SiT-B/2	×	×	38.60	41.95
V1	✓	×	32.20	52.38
V2	×	✓	25.21	63.83
Ours	✓	✓	**22.91**	**70.37**

Hyperparameters. We ablate the guiding model, loss weight λ_{guide}, guidance scale ω, and the choice of guiding/guided layers, as shown in the Table 5. The top row is our default setting (layer 6 guided by layer 8 with $\omega = 30.0$).

(1) Guiding and guided layers. We evaluate different layer pairs as shown in Table 5. '$m \rightarrow n$' means that the m^{th} layer of the guiding model is used to supervise the n^{th} layer of the current model. Guiding shallow layers (*e.g.*, 8→6 and 8→4) consistently outperforms guiding deep layers (8→8), likely because constraining layers close to the output limits adaptation. Guiding the same layer depth (6→6) results in noticeably worse performance, suggesting a small abstraction gap is needed. Using very deep guiding layers (10→6) slightly under-performs 8→6, indicating overly semantic features are less effective for shallow layers (as also noted in REPA [51]). Overall, good guidance requires features that are semantically strong yet well-aligned with the target layer.

Table 5: Ablation studies on the guidance scale, guiding and guided layers, the selection of the guiding model, and the loss weight λ_{guide}. '$m \rightarrow n$' means the n^{th} layer of DiT model is guided by the m^{th} layer of guiding model.

Block Layers	Guidance Scale	Guiding model, Training iters	λ_{guide}	FID↓	IS↑
8 → 6	30.0	$M_{align.}$, 200K	0.5	22.91	70.37
8 → 4	30.0	$M_{align.}$, 200K	0.5	23.43	70.25
8 → 8	30.0	$M_{align.}$, 200K	0.5	24.29	66.88
6 → 6	30.0	$M_{align.}$, 200K	0.5	24.37	67.89
10 → 6	30.0	$M_{align.}$, 200K	0.5	24.05	67.94
8 → 6	1.0	$M_{align.}$, 200K	0.5	29.30	55.68
8 → 6	15.0	$M_{align.}$, 200K	0.5	24.26	68.36
8 → 6	45.0	$M_{align.}$, 200K	0.5	23.01	70.69
8 → 6	60.0	$M_{align.}$, 200K	0.5	23.57	68.20
8 → 6	30.0	$M_{align.}$, 50K	0.5	28.32	57.22
8 → 6	30.0	$M_{align.}$, 100K	0.5	25.20	64.05
8 → 6	30.0	$M_{align.}$, 300K	0.5	23.05	70.32
8 → 6	30.0	M_{ori}, 200K	0.5	25.21	63.83
8 → 6	30.0	M_{repa}, 200K	0.5	23.40	70.81
8 → 6	30.0	$M_{align.}$, 200K	0.1	24.28	67.50
8 → 6	30.0	$M_{align.}$, 200K	0.3	23.08	71.41
8 → 6	30.0	$M_{align.}$, 200K	0.7	23.27	70.12
8 → 6	30.0	$M_{align.}$, 200K	1.0	23.24	68.97

(2) Guidance scale. We vary the guidance scale in the self-guided stage from 1.0 to 60.0 (Table middle). A small scale (1.0) greatly hurts performance, showing that weak guidance cannot transfer semantics well. A larger scale (45.0) slightly

Table 6: Training cost comparison on SiT-B/2 with batch size 256 on ImageNet 256×256 using 8×A800 GPUs.

Method	Speed (iters/s)↑	Total time (h)↓	Peak memory (GB/card)↓
Vanilla (SiT) [26]	9.28	58.87	9.03
REPA [51]	8.49	65.44	10.39
Self-Transcendence	Stage-1: 6.69 (first 50K iters) Stage-2: 9.28 (remaining 1950K iters)	60.45	12.34 (during first 50K iters) 9.03 (after early stop)

improves IS but does not outperform the default setting. An extreme scale (60.0) degrades both FID and IS, likely due to instability or overfitting to intermediate features [5, 28]. Overall, a moderate scale (*i.e.*, 30.0) best balances gains.

(3) Guiding model. Thirdly, we study the effect of the guiding model. Models trained with VAE structure guidance, vanilla diffusion loss, and REPA are denoted as M_{align}, M_{ori}, and M_{repa}, respectively. **Training length.** We train M_{align} for 50K–300K iterations. Better-trained teachers improve FID (*e.g.*, 28.32 at 50K vs. 22.91 at 200K), but 300K brings no further gain (FID 23.05). This implies that around 200K steps, the intermediate features of the guiding model become semantically rich enough to offer useful supervision. With further training, its internal representations may shift from the current model. This is similar to the representation drift problem observed in knowledge distillation [5, 28], where a too-strong teacher may misguide the student. **Teacher type.** we compare different types of guiding models. Using a model trained with standard diffusion loss (M_{ori}) results in worse performance, showing that the lack of VAE structure guidance reduces guiding quality. Meanwhile, using a model trained with REPA (M_{repa}) with a stronger performance also achieves worse performance than ours, suggesting our carefully designed guiding model $M_{align.}$ provides both the more effective structural and semantic priors.

(4) Loss weight. We ablate the loss weights in Self-Transcendence. Training uses diffusion loss $\mathcal{L}_{diff}$ and the self-guided loss $\mathcal{L}_{guide}$, with $\lambda_{diff} = 1.0$ and $\lambda_{guide} \in [0.1, 1.0]$ (Table 5). The best result is at $\lambda_{guide} = 0.5$ (FID 22.91, IS 70.37). Too small $\mathcal{L}_{guide}$ weakens guidance ($\lambda_{guide} = 0.1$: FID 24.28), while too large λ_{guide} hurts performance, likely due to over-regularization ($\lambda_{guide} = 1.0$: FID 23.24). These results confirm the importance of balancing the two loss terms. A moderate guidance weight (*e.g.*, 0.5) provides optimal control without overwhelming the primary diffusion objective. Therefore, in our final model, we choose $\lambda_{guide} = 0.5$ as the default setting.

4.4 Training Cost Comparison

We compare the training cost of Self-Transcendence, REPA [51], and the vanilla model [26] in terms of both time and GPU memory in Table 6. All models are trained with batch size 256. Using SiT-B/2 as an example, we train for 400 epochs (2,000K iterations) on 8 A800 GPUs. The vanilla model runs at 9.28 iters/s, taking about 58.87 hours in total. Its peak GPU memory is about 9.03 GB per card. With REPA, the speed drops to 8.49 iters/s, resulting in 65.44 hours. REPA relies on a pretrained DINOv2 model [31], which brings extra cost

and increases the peak memory to 10.39 GB. For Self-Transcendence, the first 50K iterations run at 6.69 iters/s, and the remaining 1950K iterations run much faster, reaching 9.28 iters/s, leading to 60.45 hours in total. In terms of memory, our method only adds a guiding model in the early stage, whose peak memory is 12.34 GB during the first 50K iterations, and then returns to the vanilla level (9.03 GB) after early stop.

Table 7: Comparisons on linear probing on ImageNet classification using different DiT layers features.

Layers	4	8	12	16	20	24
SiT-XL/2	11.19%	11.87%	16.25%	29.49%	42.15%	39.13%
+ REPA	26.47%	51.87%	52.07%	49.76%	47.98%	44.52%
+ Ours	25.06%	41.18%	49.13%	50.00%	47.46%	43.52%

Overall, our method has a comparable training time to the vanilla model and is faster than REPA. It also has a smaller memory overhead than REPA, and keeps the long-run memory cost close to vanilla. Moreover, training our guiding model takes only 6.39 hours (200K iterations at 8.70 iters/s), which is substantially cheaper than pretraining a large teacher such as DINOv2.

4.5 Understanding Capability.

To show the improvement on understanding capability, we evaluate the features at $t = 0.5$ by linear probing on ImageNet classification in Table 7. Self-transcendence clearly improves vanilla SiT-XL with enhanced discriminative capability, though it stays below REPA, which uses a DINO-pretrained teacher [31]. Notably, stronger linear separability does not necessarily mean better generation: our guiding features share the same diffusion setting as the student and are thus better aligned with the generative objective, explaining our superior generation.

5 Conclusion and Limitation

We proposed **Self-Transcendence**, a simple yet effective self-guided training framework to accelerate the training of diffusion transformers (DiTs). Unlike previous approaches that relied on external models for semantic supervision, our method was entirely self-contained by leveraging the model's own internal features to guide its training. We designed a two-stage pipeline, where we first aligned shallow-layer features with VAE latents to provide stable early supervision and then applied classifier-free guidance to enhance the semantic expressiveness of intermediate features. The obtained features were used to guide a new DiT training. Extensive experiments demonstrated that our method achieved comparable or even superior performance to externally guided methods such as REPA. Our findings highlighted the untapped potential of internal representations in DiT models and provided a new direction for self-supervised acceleration.

Limitations. While our method eliminates the need for external models, the quality of internal guidance is still upper-bounded by the model's own capacity. Secondly, our method introduces additional hyperparameters, which may require tuning across tasks. Finally, as in previous works [13, 44, 45, 50], our approach is evaluated on image generation benchmarks; its applicability to other generative modalities (*e.g.*, text-to-video generation and text-to-3D generation) deserves to be explored in future work.

References

1. Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics **2**(4), 433–459 (2010). https://doi.org/10.1002/wics.101
2. Anonymous: Dense2moe: Unifying pruning and upcycling for efficient large language models. In: Submitted to The Fourteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=hYGPetyGSr, under review
3. Bao, F., Li, C., Cao, Y., Zhu, J.: All are worth words: a vit backbone for score-based diffusion models. In: NeurIPS 2022 Workshop on Score-Based Methods (2022), https://openreview.net/forum?id=WfkBiPO5dsG
4. Chen, H., Han, Y., Chen, F., Li, X., Wang, Y., Wang, J., Wang, Z., Liu, Z., Zou, D., Raj, B.: Masked autoencoders are effective tokenizers for diffusion models. In: International Conference on Learning Representations, year=2025,
5. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4793–4801 (2019). https://doi.org/10.1109/ICCV.2019.00489
6. Clark, K., Jaini, P.: Text-to-image diffusion models are zero shot classifiers. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), https://openreview.net/forum?id=fxNQJVMwK2
7. Dai, W., Li, J., Li, D., Tiong, A., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), https://openreview.net/forum?id=vvoWPYqZJA
8. Deng, C., Zhu, D., Li, K., Gou, C., Li, F., Wang, Z., Zhong, S., Yu, W., Nie, X., Song, Z., Shi, G., Fan, H.: Emerging properties in unified multimodal pretraining (2025), https://arxiv.org/abs/2505.14683
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy
10. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first International Conference on Machine Learning (2024), https://openreview.net/forum?id=FPnUhsQJ5B
11. Gao, S., Zhou, P., Cheng, M.M., Yan, S.: Masked diffusion transformer is a strong image synthesizer (2023)
12. Gao, S., Zhou, P., Cheng, M.M., Yan, S.: Mdtv2: Masked diffusion transformer is a strong image synthesizer (2024), https://arxiv.org/abs/2303.14389
13. Haghighi, Y., van Delft, B., Hassan, M., Alahi, A.: Layersync: Self-aligning intermediate layers (2025), https://arxiv.org/abs/2510.12581
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
15. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
16. Jiang, D., Wang, M., Li, L., Zhang, L., Wang, H., Wei, W., Dai, G., Zhang, Y., Wang, J.: No other representation component is needed: Diffusion transformers can

provide representation guidance by themselves. arXiv preprint arXiv:2505.02831 (2025)
17. Krause, F., Phan, T., Gui, M., Baumann, S.A., Hu, V.T., Ommer, B.: Tread: Token routing for efficient architecture-agnostic diffusion training. arXiv preprint arXiv:2501.04765 (2025)
18. Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. Advances in neural information processing systems **32** (2019)
19. Labs, B.F.: Flux. `https://github.com/black-forest-labs/flux` (2024), official inference repo for FLUX.1 models
20. Lee, H.Y., Tseng, H.Y., Lee, H.Y., Yang, M.H.: Exploiting diffusion prior for generalizable dense prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
21. Leng, X., Singh, J., Hou, Y., Xing, Z., Xie, S., Zheng, L.: Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers (2025)
22. Li, A.C., Prabhudesai, M., Duggal, S., Brown, E.L., Pathak, D.: Your diffusion model is secretly a zero-shot classifier. In: ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling (2023), `https://openreview.net/forum?id=Ck3yXRdQXD`
23. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., Li, C.: LLaVA-onevision: Easy visual task transfer. Transactions on Machine Learning Research (2025), `https://openreview.net/forum?id=zKv8qULV6n`
24. Li, T., Katabi, D., He, K.: Return of unconditional generation: A self-supervised representation generation method. Advances in Neural Information Processing Systems **37**, 125441–125468 (2024)
25. Ma, G., Huang, H., Yan, K., Chen, L., Duan, N., Yin, S., Wan, C., Ming, R., Song, X., Chen, X., Zhou, Y., Sun, D., Zhou, D., Zhou, J., Tan, K., An, K., Chen, M., Ji, W., Wu, Q., Sun, W., Han, X., Wei, Y., Ge, Z., Li, A., Wang, B., Huang, B., Wang, B., Li, B., Miao, C., Xu, C., Wu, C., Yu, C., Shi, D., Hu, D., Liu, E., Yu, G., Yang, G., Huang, G., Yan, G., Feng, H., Nie, H., Jia, H., Hu, H., Chen, H., Yan, H., Wang, H., Guo, H., Xiong, H., Xiong, H., Gong, J., Wu, J., Wu, J., Wu, J., Yang, J., Liu, J., Li, J., Zhang, J., Guo, J., Lin, J., Li, K., Liu, L., Xia, L., Zhao, L., Tan, L., Huang, L., Shi, L., Li, M., Li, M., Cheng, M., Wang, N., Chen, Q., He, Q., Liang, Q., Sun, Q., Sun, R., Wang, R., Pang, S., Yang, S., Liu, S., Liu, S., Gao, S., Cao, T., Wang, T., Ming, W., He, W., Zhao, X., Zhang, X., Zeng, X., Liu, X., Yang, X., Dai, Y., Yu, Y., Li, Y., Deng, Y., Wang, Y., Wang, Y., Lu, Y., Chen, Y., Luo, Y., Luo, Y., Yin, Y., Feng, Y., Yang, Y., Tang, Z., Zhang, Z., Yang, Z., Jiao, B., Chen, J., Li, J., Zhou, S., Zhang, X., Zhang, X., Zhu, Y., Shum, H.Y., Jiang, D.: Step-video-t2v technical report: The practice, challenges, and future of video foundation model (2025), `https://arxiv.org/abs/2502.10248`
26. Ma, N., Goldstein, M., Albergo, M.S., Boffi, N.M., Vanden-Eijnden, E., Xie, S.: Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers (2024), `https://arxiv.org/abs/2401.08740`
27. Meng, B., Xu, Q., Wang, Z., Cao, X., Huang, Q.: Not all diffusion model activations have been evaluated as discriminative features. Advances in Neural Information Processing Systems **37**, 55141–55177 (2024)
28. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 5191–5198 (2020)
29. Nash, C., Menick, J., Dieleman, S., Battaglia, P.W.: Generating images with sparse representations. arXiv preprint arXiv:2103.03841 (2021)

30. OpenAI: Video generation models as world simulators. https://openai.com/index/video-generation-models-as-world-simulators/ (2024)
31. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., HAZIZA, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research (2024), https://openreview.net/forum?id=a68SUt6zFt, featured Certification
32. Peebles, W., Xie, S.: Scalable diffusion models with transformers (2023), https://arxiv.org/abs/2212.09748
33. Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., Qiu, Z.: Qwen2.5 technical report (2025), https://arxiv.org/abs/2412.15115
34. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10684–10695 (June 2022)
35. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Advances in neural information processing systems **29** (2016)
36. Shazeer, N.: Glu variants improve transformer (2020), https://arxiv.org/abs/2002.05202
37. Singh, J., Leng, X., Wu, Z., Zheng, L., Zhang, R., Shechtman, E., Xie, S.: What matters for representation alignment: Global information or spatial structure? arXiv preprint arXiv:2512.10794 (2025)
38. Su, J., Lu, Y., Pan, S., Wen, B., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding (2021)
39. Tian, Y., Chen, H., Zheng, M., Liang, Y., Xu, C., Wang, Y.: U-repa: Aligning diffusion u-nets to vits (2025), https://arxiv.org/abs/2503.18414
40. Tian, Y., Tu, Z., Chen, H., Hu, J., Xu, C., Wang, Y.: U-dits: Downsample tokens in u-shaped diffusion transformers. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems (2024), https://openreview.net/forum?id=SRWs2wxNs7
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
42. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., Zeng, J., Wang, J., Zhang, J., Zhou, J., Wang, J., Chen, J., Zhu, K., Zhao, K., Yan, K., Huang, L., Feng, M., Zhang, N., Li, P., Wu, P., Chu, R., Feng, R., Zhang, S., Sun, S., Fang, T., Wang, T., Gui, T., Weng, T., Shen, T., Lin, W., Wang, W., Wang, W., Zhou, W., Wang, W., Shen, W., Yu, W., Shi, X., Huang, X., Xu, X., Kou, Y., Lv, Y., Li, Y., Liu, Y., Wang, Y., Zhang, Y., Huang, Y., Li, Y., Wu, Y., Liu, Y., Pan, Y., Zheng, Y., Hong, Y., Shi, Y., Feng, Y., Jiang, Z.,

Han, Z., Wu, Z.F., Liu, Z.: Wan: Open and advanced large-scale video generative models (2025), https://arxiv.org/abs/2503.20314
43. Wang, C., Zhou, C., Gupta, S., Lin, Z., Jegelka, S., Bates, S., Jaakkola, T.: Learning diffusion models with flexible representation guidance. arXiv preprint arXiv:2507.08980 (2025)
44. Wang, R., He, K.: Diffuse and disperse: Image generation with representation regularization (2025), https://arxiv.org/abs/2506.09027
45. Wang, S., Tian, Z., Huang, W., Wang, L.: Ddt: Decoupled diffusion transformer. arXiv preprint arXiv:2504.05741 (2025)
46. Wang, Z., Zhao, W., Zhou, Y., Li, Z., Liang, Z., Shi, M., Zhao, X., Zhou, P., Zhang, K., Wang, Z., et al.: Repa works until it doesn't: Early-stopped, holistic alignment supercharges diffusion training. arXiv preprint arXiv:2505.16792 (2025)
47. Wu, G., Zhang, S., Shi, R., Gao, S., Chen, Z., Wang, L., Chen, Z., Gao, H., Tang, Y., Yang, J., Cheng, M.M., Li, X.: Representation entanglement for generation: Training diffusion transformers is much easier than you think (2025), https://arxiv.org/abs/2507.01467
48. Xie, E., Chen, J., Chen, J., Cai, H., Tang, H., Lin, Y., Zhang, Z., Li, M., Zhu, L., Lu, Y., Han, S.: Sana: Efficient high-resolution image synthesis with linear diffusion transformer (2024), https://arxiv.org/abs/2410.10629
49. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Yuxuan.Zhang, Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., Tang, J.: Cogvideox: Text-to-video diffusion models with an expert transformer. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=LQzN6TRFg9
50. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
51. Yu, S., Kwak, S., Jang, H., Jeong, J., Huang, J., Shin, J., Xie, S.: Representation alignment for generation: Training diffusion transformers is easier than you think. In: International Conference on Learning Representations (2025)
52. Zhang, B., Sennrich, R.: Root Mean Square Layer Normalization. In: Advances in Neural Information Processing Systems 32. Vancouver, Canada (2019), https://openreview.net/references/pdf?id=S1qBAf6rr
53. Zhao, W., Han, Y., Tang, J., Wang, K., Song, Y., Huang, G., Wang, F., You, Y.: Dynamic diffusion transformer. arXiv preprint arXiv:2410.03456 (2024)
54. Zheng, H., Nie, W., Vahdat, A., Anandkumar, A.: Fast training of diffusion models with masked transformers. Transactions on Machine Learning Research (2024), https://openreview.net/forum?id=vTBjBtGioE
55. Zhu, R., Pan, Y., Li, Y., Yao, T., Sun, Z., Mei, T., Chen, C.W.: Sd-dit: Unleashing the power of self-supervised discrimination in diffusion transformer (2024), https://arxiv.org/abs/2403.17004
56. Zhu, R., Pan, Y., Li, Y., Yao, T., Sun, Z., Mei, T., Chen, C.W.: Sd-dit: Unleashing the power of self-supervised discrimination in diffusion transformer (2024), https://arxiv.org/abs/2403.17004