Unifying CNNs and ViTs for Learning-Efficient and Scalable Variational AutoEncoder

Zhiying Lu[1], Shang Chai[2], Chuanbin Liu*[1], Litong Gong[2], Pandeng Li[1], Tiezheng Ge[2], and Hongtao Xie[1]

[1] University of Science and Technology of China, China
[2] Taobao & Tmall Group of Alibaba, China
{arieseirack,lpd}@mail.ustc.edu.cn
{liucb92,htxie}@ustc.edu.cn
{chaishang.cs,gonglitong.glt,tiezheng.gtz}@taobao.com

Abstract. Continuous variational autoencoders (VAEs) are widely used as visual tokenizers for latent diffusion models. While CNN-based VAEs reliably capture local details, they model long-range dependencies inefficiently. Vision Transformers (ViTs) provide global context, but existing ViT-VAEs exhibit three practical limitations in this setting: (i) weak resolution extrapolation when trained at a single resolution, (ii) slow convergence with suboptimal fine-grained color and texture fidelity, and (iii) inconsistent scaling gains when increasing model size. To address these challenges, we propose **TransVAE**, a hybrid CNN-ViT VAE that combines a shallow CNN front-end for local feature extraction with a deep Transformer backbone for global modeling. We apply minimal yet principled architectural modifications and systematically analyze their effectiveness, including a pure Rotary Position Embedding (RoPE) strategy, a multi-stage design, and a convolutional feed-forward network. Across model sizes from 44M to 2.3B parameters, TransVAE shows learning-efficient training and predictable scaling improvements, and it supports single-resolution training (256×256) with robust inference at higher resolutions ($512\times512/1024\times1024$). Moreover, TransVAE achieves competitive reconstruction and downstream generation performance compared to strong tokenizers, such as FLUX-VAE and VA-VAE, offering a favorable trade-off between reconstruction fidelity and generation-friendly latents.

Keywords: VAE · Vision Transformer · Image Generation

1 Introduction

Since the advent of latent diffusion models [20, 21], modern generative models rely on visual tokenizers to reduce the computational cost by compressing large-resolution visual inputs to a low-dimensional latent space. Among different tokenizers, the continuous-value variational autoencoder (VAE) [11] stands out, as it enables high-fidelity encoding and decoding of visual information. For years,

* Corresponding Author

(a) Reconstruction Visualization of different VAEs in Early Training Phase

(b) Convergence Curve of Different VAEs

Fig. 1: Comparison of the early patterns and loss curves of CNN-VAE, ViT-VAE, and our proposed TransVAE. TransVAE learns both fine-grained details and global structure early, leading to faster convergence.

VAE architectures have been dominated by CNNs [7, 13, 21]. This paradigm has been successful, leveraging strong inductive biases such as locality and translation equivariance to efficiently learn hierarchical features from pixel-level data [10, 12, 25, 26]. Despite their success, CNNs face a fundamental limitation: the local nature of the convolution operator makes it inefficient at modeling long-range, non-local dependencies. Capturing the global structure of a scene often requires deep networks, leading to computational and optimization challenges.

The rise of Vision Transformers (ViTs) presents a complementary alternative, offering a native mechanism for global context modeling through their self-attention layers [6]. Yet, in the continuous VAE tokenizer setting, existing ViT-VAEs exhibit three practical limitations that hinder their adoption:

1. **Weak resolution extrapolation.** Models trained at a single resolution often fail to generalize reliably to unseen higher resolutions during inference, largely due to how spatial information is represented. For example, absolute positional embeddings require interpolation at unseen resolutions, which can break learned spatial priors.
2. **Slow convergence and suboptimal local fidelity.** Without strong local inductive biases, ViT-VAEs tend to learn fine-grained color and texture details inefficiently, leading to slower convergence and visible artifacts, especially in early training, as shown in Fig. 1.
3. **Inconsistent scaling gains.** Increasing parameter count in pure ViT-VAEs does not consistently translate into improved reconstruction quality, limiting their scalability as visual tokenizers.

These limitations point to an architectural gap: a purely local CNN or purely global ViT alone often captures an *incomplete* representation for high-fidelity VAEs. This motivates a principled unification for complementary strengths.

To address the above challenges, we propose **TransVAE**, a hybrid CNN-ViT VAE architecture that combines a shallow CNN front-end for robust local feature extraction with a deep Transformer backbone for global context modeling. TransVAE is built with minimal yet principled modifications that directly target the above issues: (i) a RoPE-only positional encoding [23] for robust resolution extrapolation; (ii) a multi-stage design with a CNN stem to improve learning

efficiency for fine-grained details; and (iii) a convolutional feed-forward network to inject local inductive bias into deep Transformer blocks. Together, this unification yields a more complete representation and enables predictable scaling benefits.

Extensive experiments validate the effectiveness of TransVAE. Across a family of models from 44M to 2.3B parameters, TransVAE demonstrates learning-efficient training and predictable scaling improvements. To be specific, VA-VAE achieves 28.57 PSNR at 100 epochs, while under the same setting, TransVAE achieves 28.61 PSNR at 25 epochs and reaches 29.08 PSNR at 100 epochs. Moreover, TransVAE supports single-resolution training (256×256) while performing robust inference at higher resolutions (512×512 / 1024×1024). Finally, TransVAE achieves competitive reconstruction and downstream generation performance compared to strong recent tokenizers, such as FLUX-VAE [13] and VA-VAE [31], offering a favorable trade-off between reconstruction fidelity and generation-friendly latents.

Our contributions can be summarized as follows:

- We identify three practical limitations of existing ViT-VAEs as continuous visual tokenizers: resolution extrapolation, learning efficiency for fine-grained details, and inconsistent scaling benefits.
- We propose **TransVAE**, a hybrid CNN–ViT VAE with minimal, principled modifications (RoPE-only, multi-stage CNN-Transformer design, and Convolutional FFN) that directly address these limitations.
- We scale TransVAE from 44M to 2.3B parameters and demonstrate predictable improvements on reconstruction, semantic representation alignment, and downstream generation together with robust resolution extrapolation.

2 Related Work

Tokenizers. Since the advent of Latent Diffusion Models (LDM) [20,21], VAE has become the core component to accelerate training and inference by compressing the image from a large spatial scale to a compressed latent space. The widespread VAEs [7, 13, 21] are built upon CNN structures, leveraging the local prior to compress and extract detailed visual representations, and reconstructing the latents with the decoder with a symmetric structure. Though the ViT [6] structure has been widely adopted in discrete 1D tokenizers [1, 2, 18, 32], the ViT structure for the continuous variational autoencoder has not been fully explored. ViTok [9] is the first to study the potential, but it fails in scaling and falls behind traditional CNN-VAEs in reconstruction quality. In addition, DeTok [29] and MAGI-VAE [24] also underperform normal CNN-VAEs and fail to extrapolate to other resolutions during inference, as shown in Fig. 3. In this paper, we focus on the architecture design of VAE with a hybrid CNN-ViT structure, breaking the common CNN architecture and overcoming the limitations of ViT architectures. With its more complete visual representations, our TransVAE demonstrates superior performance in learning efficiency and scalability.

Hybrid CNN-ViT. The vanilla ViT requires substantial training cost to converge. Thus, researchers develop a series of hybrid structure models [16, 17, 27, 28, 34] combining CNN and ViT, leveraging the local prior to facilitate learning

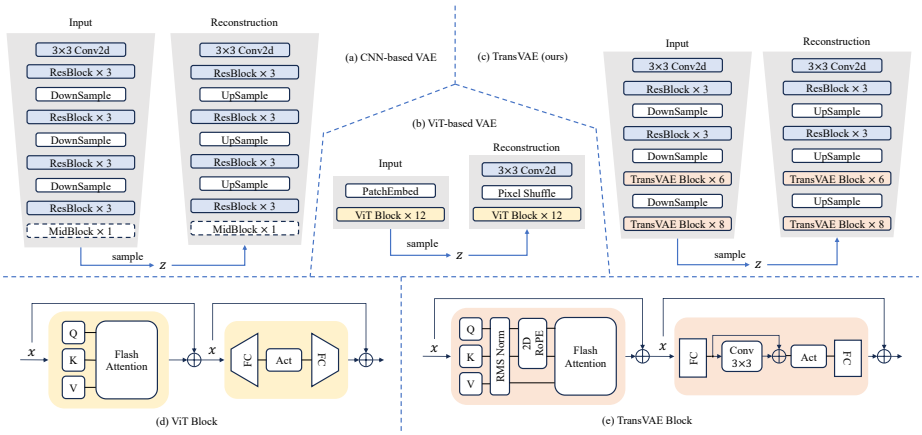


Fig. 2: Comparison of CNN-VAE, ViT-VAE, and our TransVAE. TransVAE uses a CNN stem for patch embedding and fine-grained pixel representation, followed by stages of TransVAE blocks to capture global and semantic representations. The TransVAE block is equipped with normalization to stabilize the scaling-up training, RoPE to enable resolution extrapolation, and a ConvFFN with residual convolution to obtain local bias in the deep network.

efficiency. CeiT [34] and CvT [28] leverage convolution for patch embedding, replacing the non-overlapping patch embedding in vanilla ViT. PVT [27] redesigns the structure of vanilla ViT into a hierarchical style. DHVT [16, 17] further integrates depth-wise convolution into the feed-forward network. However, the hybrid CNN-ViT structure is largely underexplored in VAEs. The existing DC-AE [4] is the first to adopt a hybrid structure, while it adopts many convolution blocks in early stages and only a few ViT blocks in late stages. Thus, it is different from our TransVAE as we focus more on TransVAE blocks in the late stages.

3 Method

A core tenet of our design philosophy is to maintain **architectural simplicity** to facilitate scalability. In this section, we elaborate on the TransVAE architecture with experiments to show the effectiveness of the proposed components. First, we introduce the preliminaries about the structure and early learning patterns of traditional CNN-VAE and the ViT-VAE. In addition, we conduct comprehensive experiments to explore how each component contributes to the learning efficiency, gradually strengthening ViT-VAE to our TransVAE, which achieves a visual representation with local and global modeling capacity. Finally, with this more complete representation, we break the scaling limitation of traditional VAEs of different architectures, demonstrating a consistent scaling benefit on reconstruction and alignment.

3.1 Preliminaries

Background We focus primarily on continuous image VAEs, as they minimize information loss within the compressed latent space and consistently outperform discrete counterparts in modern generative models. A VAE consists of two principal components: an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. Given an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$, the encoder $\mathcal{E}$ maps the high-dimensional $\mathbf{x}$ to a lower-dimensional latent representation. By employing Gaussian modeling and Kullback-Leibler (KL) regularization, the VAE constructs an information-rich compressed latent space $\mathbf{z} = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{\frac{H}{f}\times\frac{W}{f}\times d}$, where f and d denote the spatial compression ratio and latent dimension, respectively. The decoder then reconstructs the latent representation back to the original signal space as $\hat{\mathbf{x}} = \mathcal{D}(\mathbf{z})$ by minimizing the reconstruction loss with respect to $\mathbf{x}$.

Historically, CNNs have dominated the architectural design of VAE encoders and decoders, as illustrated in Fig. 2(a). Standard CNN-VAEs utilize Residual Blocks (ResBlocks) from ResNet [10] to construct each stage, employing downsampling and upsampling operations across stages. Some variants [7, 13, 21] incorporate a MidBlock, typically comprising sandwich-like Conv-Attn-Conv layers between the final convolution stage and the latent representation to provide global modeling capabilities. However, as discussed in the Supplementary Materials, this attention module often proves less effective during training. Thus, we focus on pure CNN-VAEs without MidBlock designs.

Preliminary Analysis We first introduce the experimental settings to train VAEs with $f16d32$ for all different structures. We adopt the ImageNet-1k datasets at 256×256 resolution as the training dataset, with a batch size of 32, and train all models for 5 epochs, resulting in 200k steps in total. The loss curves are averaged over three random seeds. We only focus on the reconstruction pattern in this Sec. 3, which means our training target is L1 loss for reconstruction, LPIPS loss [36], and KL loss, respectively. We keep the 5 epochs training for all analyses in Sec. 3 and Fig. 1 to show the early training behavior, except that models in Fig. 3 are trained for 100 epochs, demonstrating the final convergence performance. More details are shown in the Supplementary Materials.

Fig. 1 visualizes the behavior of CNN-VAEs and ViT-VAEs during the early learning stages of image reconstruction. Two key observations emerge: (1) CNN-VAEs excel at fine-grained local modeling, such as capturing pixel-level color, whereas ViT-VAEs capture coarse-grained global representations, such as shape and structure. (2) While CNN-VAEs gradually learn global representations, they often fail to capture structural details, whereas ViT-VAEs struggle with pixel-level color fidelity and exhibit slower convergence. We attribute these phenomena to the incomplete visual representations inherent to each architecture when used in isolation. By hybridizing these structures, our proposed TransVAE captures both local and global representations early in training, converging faster than CNN-VAEs and eliminating structural artifacts. The TransVAE architecture, shown in Fig. 2, features a multi-stage hybrid design, Rotary Position Embeddings (RoPE)

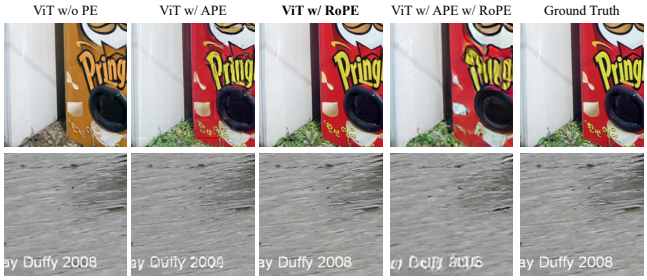


Fig. 3: Visualization of extrapolation inference ($256 \rightarrow 512$) for different types of models trained for 100 epochs, demonstrating the final convergence performance.

Table 1: Position Embedding of ViT-VAEs. MAGI-VAE, DeTok, and SoftVQ apply APE after patch embedding, while MAETok adopts APE in the latent space. Our TransVAE applies RoPE-only design.

APE	RoPE	Existing ViT Tokenizers
✗	✗	None
✓	✗	DeTok [29], SoftVQ [2]
✗	✓	TransVAE (Ours)
✓	✓	MAETok [1], MAGI-VAE [24]

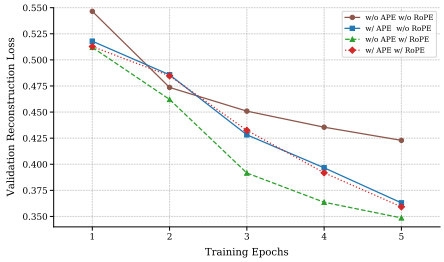


Fig. 4: Loss Curve on Position Embedding. Models are trained for 5 epochs.

for resolution extrapolation, normalization for stable scaling, and a Convolutional Feed-Forward Network (ConvFFN) to further enhance local priors. We detail the contribution of each component in the following sections.

3.2 Macro Design

We divide our improvements into two levels: macro designs and micro designs. The macro design includes Position Embedding strategy, Patch Embedding module, and Architectural Staging. The micro design includes the modifications in the Transformer Block for Attention and Feed-Forward Network (FFN), and the Upsample or Downsample Module.

Resolution Extrapolation with RoPE Modern VAEs are increasingly required to generalize to resolutions not encountered during training. This capability, termed extrapolation, depends heavily on the model's mechanism for handling spatial information. The Position Embedding (PE) strategy is a critical components for ViT-VAEs considering extrapolation. Existing ViT methods apply different PEs, as shown in Tab. 1, while some of them fall short in extrapolation inference. We investigate this by training ViT-VAEs for 100 epochs with different

PE, including Absolute Position Embeddings (APE) and Rotary Position Embeddings (RoPE) [23], on a fixed resolution of 256×256 and evaluating them directly on a higher resolution of 512×512. As shown in Fig. 3, once APE is applied, models exhibit a critical failure mode. APE is a learnable parameter matrix $\mathbf{P}_{APE} \in \mathbb{R}^{N \times D}$, where N is the sequence length and D is the feature dimension. APE is added to the token embeddings $\mathbf{X} \in \mathbb{R}^{N \times D}$ by $\mathbf{X} = \mathbf{X} + \mathbf{P}_{APE}$. When the inference sequence length differs from the training length, $\mathbf{P}_{APE}$ must be interpolated. This process disrupts the learned spatial priors, resulting in grid-like artifacts and degraded reconstruction quality.

To address this limitation, we replace APE with RoPE in the self-attention module. Instead of adding static positional information, RoPE applies a rotation g to the query $\mathbf{q}_m$ and key $\mathbf{k}_n$ vectors based on their absolute positions m and n. The transformed vectors are defined as $\mathbf{q}_m^{'} = g(\mathbf{q}_m, m)$ and $\mathbf{k}_n^{'} = g(\mathbf{k}_n, n)$. Consequently, their inner product becomes $(\mathbf{q}_m')^\top \mathbf{k}_n' = h(\mathbf{q}_m, \mathbf{k}_n, m-n)$. The crucial insight is that the attention score depends on the relative distance $m - n$ rather than absolute positions. Because the rotation applies to any position, RoPE naturally generalizes to arbitrary sequence lengths without interpolation, thereby preserving spatial relationships across resolutions.

In addition, from the final behaviour in Fig. 3 and the early training loss curve in Fig. 4, we find that ViT-VAE without PE fails in modeling pixel-level color detail, and it is hard to converge. Applying only RoPE achieves the optimal results. It not only avoids extrapolation failures but also demonstrates better learning efficiency in reconstruction fidelity. This analysis firmly establishes RoPE as a crucial component for building a truly generalizable ViT-VAE, and thus it is a core element of our TransVAE.

Fine-grained Patch Embedding and Multi-Stage While RoPE effectively addresses the challenge of positional generalization, the fundamental weakness of ViTs in modeling fine-grained local details persists, as it is orthogonal to the position embedding strategy. We first explore the architecture of typical ViT-VAEs inherited from the original ViT for visual understanding. Standard ViT-VAEs [9, 24, 29] often employ a non-overlapping convolutional layer for patch embedding and a pixel-shuffle operation for the final output to recover the input spatial resolution, as shown in Fig. 2(b). Such designs, while computationally efficient, tend to treat local patches as independent units or merely shuffle pixels into channel dimensions, largely ignoring the local spatial correlations.

Therefore, we aim to enhance local interactions. Inspired by DHVT [17] that bridges the gap between CNN and ViT in visual understanding, we replace the standard non-overlapping patch embedding and pixel-shuffle layers with Sequential Overlapping Convolutional layers (SeqConv) for downsampling and upsampling. Specifically, as shown in Fig. 5, for the compression ratio $f = 2^k$, SeqConv repeatedly applies k times 3×3 convolution operation with a stride of 2 to downsample or upsample features. This design ensures that each downsampling/upsampling step involves a receptive field that overlaps with its neighbors, forcing the model to learn and preserve local spatial structures. The

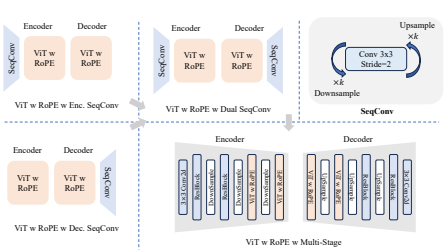


Fig. 5: Illustration of the evolution of Patch Embedding.

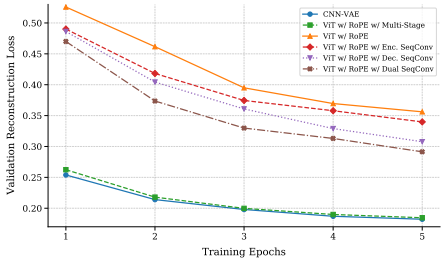


Fig. 6: Loss Curve on Patch Embedding. Models are trained for 5 epochs.

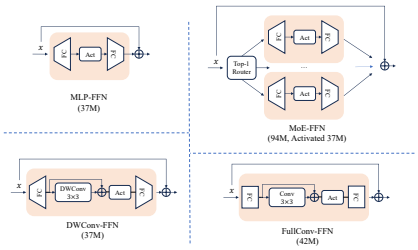


Fig. 7: Different FFN variants.

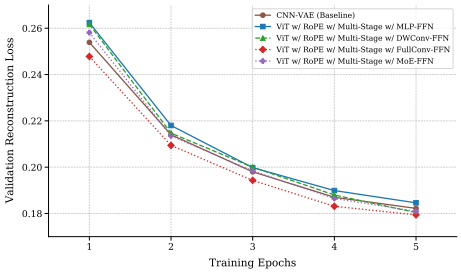


Fig. 8: Loss Curve on FFN variants.

efficacy of this approach is demonstrated in Fig. 6, where each application of our sequential convolutional layers yields an acceleration in convergence, confirming their direct contribution to more efficient local detail modeling. Applying SeqConv both for the Encoder and the Decoder greatly enhances the learning efficiency.

Furthermore, inspired by the multi-stage architecture principle in visual representation modeling, we extend this philosophy from component-level to a global architectural redesign. We transform the monolithic, single-stage ViT into a multi-stage hybrid model. Specifically, the initial two stages of our TransVAE architecture are composed of standard ResBlocks, which act as a powerful and efficient front-end for hierarchical local feature extraction. The deeper stages then apply traditional ViT blocks, resulting in the "ViT w/ RoPE w/ Multi-Stage" variant in Fig. 6, which demonstrates a close learning efficiency as CNN-VAE. This hierarchical design allows the model to first build a robust foundation of local features using the inductive biases of CNNs, before leveraging the Transformer's global attention to model high-level semantic relationships. In the next part, we then improve the ViT blocks to TransVAE blocks from a micro-level local prior enhancement perspective, thus creating a more complete visual representation.

3.3 Micro Design

Local Prior Enhancement with Convolutional FFN While the multi-stage design provides a strong foundation, a subtle performance gap exists in local detail

modeling compared to pure CNNs. We attribute this to the intrinsic structure of the Transformer block itself: its FFN operates purely point-wise on each patch token, completely severing local spatial connections at this stage of computation.

To remedy this, we introduce the convolution operation into the standard FFN design, as shown in Fig. 2(e) and Fig. 7. Specifically, we apply a convolution bypath to perform spatial mixing for the high-dimensional feature after the input projection. This core step explicitly re-establishes a local inductive bias. A residual connection is added to the convolutional path to maintain the original pointwise projected feature. We investigate two settings of convolution: a lightweight depth-wise convolution (DWConv) [22], with a $4\times$ channel expansion, inspired by DHVT [17], and a more expressive full convolution (FullConv), with a $1\times$ expansion to maintain a comparable parameter budget. To prove the effectiveness of our design, we also conduct an experiment with a Top-1 MoE-FFN variant. All the variants are illustrated in Fig. 7. The impact of our enhancement is profound. As illustrated in Fig. 8, both our variants accelerate training. The DWConv brings the convergence speed to a level that closely approaches the pure CNN baseline, while the FullConv successfully surpasses it. MoE-FFN achieves improvements at the cost of a much higher model size, while falling behind our FullConv version. This powerfully demonstrates the benefit of our local prior enhancement design, and we adopt the FullConv in our final TransVAE architecture.

Minor Modifications To complete the TransVAE design, we incorporate two minor modifications aimed at enhancing training stability at scale and preserving high-resolution fidelity for high spatial compression. First, to ensure stable training as we scale our models, we opt for RMSNorm [35] before the Q, K, and V linear projections within each self-attention block. Removing the RMSNorm may result in a NaN loss during longer training when scaling up. Second, to preserve fidelity under high compression ratios, we follow DC-AE [3] to modify all spatial downsampling and upsampling operations with a parallel, information-preserving shortcut. It will faithfully propagate information across different stages of the network. More analyses are shown in the Supplementary Materials.

These modifications, combined with the macro and micro designs discussed previously, result in our final TransVAE model. In conclusion, **we maintain architectural simplicity and effectiveness** in building TransVAE. We leave a broad design space for future work to develop more efficient architectures.

Model Configurations To demonstrate the generalization and scalability, we design TransVAE with different compression ratios and sizes. We demonstrate the model configuration, GPU memory cost, and GPU hours per epoch of TransVAE variants in Tab. 2. The Tiny and Base configurations are mainly aligned with modern VAEs: FLUX-VAE, VA-VAE, and DC-AE, for a fair comparison. The training cost is measured under BFloat16 precision with 8 NVIDIA H20 GPUs with a batch size of 32 on 256×256 resolution. The inference cost of fully-trained variants and comparisons with existing methods are shown in Sec. 4.5.

Table 2: Model configurations for TransVAE variants. We list the configuration of each Encoder stage, and the Decoder adopts the symmetric setting. The MLP ratio is always set to 1. The head dimension for self-attention is always set to 64.

Variant	Ratio	Depth	Stage Dimension	#Params			Training Cost	
				Encoder	Decoder	Total	Mem./GPU	Hours/Epoch
Tiny	$f8d16$	[3,3,3,3]	[128, 256, 512, 512]	28M	31M	60M	19G	3.5h
Large	$f8d16$	[3,3,6,8]	[192, 384, 768, 1536]	353M	366M	719M	36G	7.5h
Tiny	$f16d32$	[3,3,3,3,3]	[128, 128, 256, 256, 512]	21M	23M	44M	16G	1.5h
Base	$f16d32$	[3,3,3,3,3]	[128, 128, 256, 512, 1024]	67M	73M	140M	18G	2.5h
Large	$f16d32$	[3,3,3,4,6]	[192, 192, 384, 768, 1536]	266M	279M	545M	33G	5.5h
Huge	$f16d32$	[3,3,4,6,8]	[256, 256, 512, 1024, 2048]	633M	658M	1.3B	57G	8.5h
Giant	$f16d32$	[3,3,4,8,10]	[320, 320, 640, 1280, 2560]	1.1B	1.2B	2.3B	92G	14.5h
Base	$f32d32$	[3,3,3,3,3,3]	[128, 256, 512, 512, 1024, 1024]	131M	148M	279M	24G	4.5h

4 Experiments

We have conducted comprehensive experiments to validate the effectiveness of modules in TransVAE in Sec. 3, which can serve as ablation studies. In this section, we demonstrate the scaling properties and the performance of TransVAE, compared to modern VAEs across compression ratios and benchmarks.

4.1 Experiments Setup

(1) For reconstruction training, we follow VA-VAE [31] and apply KL regularization for the continuous latent space. We further adopt the VF loss in VA-VAE for representation alignment to obtain generation-friendly latents. We train TransVAE under different compression ratios: (i) for $f8d16$, we train two settings on an in-house large-scale dataset: multi-resolution training (w/o VF) for comparison with FLUX-VAE and SD3.5-VAE, and 256×256-only training for all VF-aligned variants; (ii) for $f16d32$ and $f32d32$, all variants are trained on ImageNet-1k at 256×256 only for fair comparison with baselines. (2) For downstream image generation, we follow VA-VAE to train LightningDiT-XL/1 and LightningDiT-B/2. All experiment settings, model configurations, latency analysis, and training data details are provided in the Supplementary Materials.

4.2 Image Reconstruction Performance

We adopt the ImageNet-1k [5] and COCO2017 [14] validation set as the main benchmarks. All models are evaluated at three resolutions. This ensures a robust evaluation of different VAEs across different data distributions and arbitrary resolutions. As shown in Tab. 3 and Tab. 4, our TransVAE achieves competitive performance compared to existing methods across multiple datasets and resolutions. To be specific, VA-VAE requires 100 epochs to reach 28.57 PSNR, while our TransVAE achieves 28.61 PSNR at 25 epochs and reaches 29.08 PSNR at 100 epochs. Both sizes of our TransVAE perform better than VA-VAE on 256×256 resolution that have been trained on, while our larger size model demonstrates

Table 3: Comparison with ViT tokenizers. **Models are trained at 256×256 only, except DC-AE.** Results are re-implemented under the same setting, except ViTok that are referred to the paper. "VF" alignment training is only on 256×256 resolution.

Tokenizer	# Params	Variant	Aligned	256 only	ImageNet-1k Val rFID ↓	PSNR ↑	LPIPS ↓	SSIM ↑	MSCOCO2017 Val rFID ↓	PSNR ↑	LPIPS ↓	SSIM ↑
256 × 256												
MAETok-B-128 [1]	176M	*f16*	✗	✓	0.48	23.61	-	0.76	4.87	23.31	-	0.77
DeTok-BB-FT [29]	171M	*f16d16*	✗	✓	0.54	**25.24**	0.137	0.71	3.94	**24.94**	0.135	0.72
ViTok S-B/16 [9]	129M	*f16d16*	✗	✓	0.50	24.36	-	0.75	3.94	24.45	-	0.76
ViTok S-L/16 [9]	427M	*f16d16*	✗	✓	**0.46**	24.74	-	**0.76**	**3.87**	24.82	-	**0.77**
VA-VAE-VF [31]	70M	*f16d32*	✓	✓	0.28	27.64	0.098	0.78	2.72	27.43	0.094	0.79
VA-VAE [31]	70M	*f16d32*	✗	✓	**0.27**	28.57	0.090	0.80	2.54	28.39	0.087	0.81
TransVAE-VF-T (Ours)	44M	*f16d32*	✓	✓	0.41	28.76	0.086	0.81	2.84	28.51	0.083	0.82
TransVAE-T (Ours)	44M	*f16d32*	✗	✓	0.38	29.08	0.083	0.83	2.55	28.82	0.080	0.83
TransVAE-VF-L (Ours)	545M	*f16d32*	✓	✓	0.37	29.19	0.082	0.82	2.70	28.98	0.079	0.83
TransVAE-L (Ours)	545M	*f16d32*	✗	✓	0.31	**29.78**	**0.075**	**0.84**	**2.39**	**29.60**	**0.072**	**0.85**
SoftVQ-B-64 [2]	176M	*f32*	✗	✓	0.61	22.97	-	**0.74**	5.16	22.86	-	**0.75**
DC-AE [4]	323M	*f32d32*	✗	✗	**0.66**	24.98	0.161	0.67	**4.80**	24.60	0.161	0.68
TransVAE-B (Ours)	279M	*f32d32*	✗	✓	0.80	**25.41**	**0.157**	0.70	5.02	**25.08**	**0.158**	0.71
512 × 512												
DeTok-BB-FT [29]	171M	*f16d16*	✗	✓	3.43	**25.22**	0.262	0.70	10.50	**24.69**	0.254	0.69
ViTok S-B/16 [9]	129M	*f16d16*	✗	✓	**0.18**	24.36	-	**0.75**	**3.94**	24.45	-	**0.76**
VA-VAE-VF [31]	70M	*f16d32*	✓	✓	0.24	29.38	0.119	0.80	2.40	28.49	0.120	0.79
VA-VAE [31]	70M	*f16d32*	✗	✓	0.12	31.05	0.095	0.84	1.55	30.12	0.097	0.83
TransVAE-VF-T (Ours)	44M	*f16d32*	✓	✓	0.26	29.98	0.108	0.83	2.56	28.83	0.146	0.79
TransVAE-T (Ours)	44M	*f16d32*	✗	✓	0.17	31.10	0.090	0.85	1.54	30.24	0.087	0.85
TransVAE-VF-L (Ours)	545M	*f16d32*	✓	✓	0.24	31.06	0.106	0.84	2.03	29.72	0.109	0.84
TransVAE-L (Ours)	545M	*f16d32*	✗	✓	**0.13**	**32.44**	**0.077**	**0.88**	**1.36**	**31.39**	**0.079**	**0.87**
DC-AE [4]	323M	*f32d32*	✗	✗	**0.20**	27.45	0.153	0.74	**3.26**	26.54	0.165	0.72
TransVAE-B (Ours)	279M	*f32d32*	✗	✓	0.30	**27.88**	**0.150**	**0.75**	3.44	**27.00**	**0.164**	**0.74**
1024 × 1024												
DeTok-BB-FT [29]	171M	*f16d16*	✗	✓	8.86	25.06	0.371	0.73	18.53	24.87	0.357	0.72
VA-VAE-VF [31]	70M	*f16d32*	✓	✓	0.37	33.61	0.116	0.89	2.10	32.66	0.121	0.88
VA-VAE [31]	70M	*f16d32*	✗	✓	0.06	36.15	0.071	0.93	0.75	35.04	0.077	0.92
TransVAE-VF-T (Ours)	44M	*f16d32*	✓	✓	0.24	33.71	0.101	0.89	1.23	33.44	0.110	0.88
TransVAE-T (Ours)	44M	*f16d32*	✗	✓	0.08	36.38	0.063	0.93	0.56	35.58	0.065	0.93
TransVAE-VF-L (Ours)	545M	*f16d32*	✓	✓	0.09	35.76	0.076	0.92	0.98	34.37	0.073	0.89
TransVAE-L (Ours)	545M	*f16d32*	✗	✓	**0.03**	**38.25**	**0.051**	**0.95**	**0.49**	**36.96**	**0.057**	**0.95**
DC-AE [4]	323M	*f32d32*	✗	✗	**0.10**	32.15	0.150	0.84	**1.83**	30.74	0.151	0.82
TransVAE-B (Ours)	279M	*f32d32*	✗	✓	0.12	**32.72**	**0.148**	**0.86**	1.94	**31.35**	**0.150**	**0.83**

superior performance in higher resolution over all the metrics, providing a scaling property on extrapolation. The static number of latent tokens in MAETok and SoftVQ hinders their extrapolation. DeTok fails to extrapolate, as it adopts absolute position embedding. ViTok can extrapolate with its RoPE design, while its reconstruction performance remains inferior due to insufficient local priors.

Our TransVAE achieves the best reconstruction performance regardless of training with VF loss alignment or not. However, all models exhibit a nearly 1.0 PSNR drop after alignment. This may come from the trade-off between high-level semantic and low-level pixel features in the latent space. VF loss also hinders extrapolation, as shown by the larger gap in reconstruction performance when increasing inference resolution. We argue that the single-resolution alignment training leads to overfitting to the semantics at one resolution and fails to generalize. This suggests that resolution-generalizable alignment may require more robust objectives or multi-resolution alignment training.

4.3 Scaling Properties of TransVAE

A good scaling property can be evidenced by the behavior of validation loss when increasing the model size. To evaluate the scaling properties of modern VAEs,

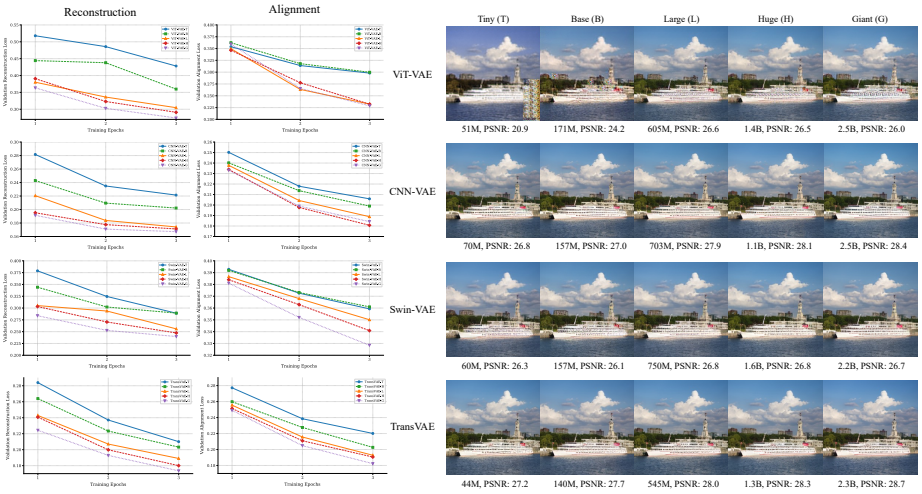
Table 4: Comparison with advanced tokenizers. Results are re-implemented under the same setting. All VF-aligned models are trained at 256 × 256 only.

Tokenizer	# Params	Variant	Aligned	256 only	ImageNet-1k Val				MSCOCO2017 Val			
					rFID ↓	PSNR ↑	LPIPS ↓	SSIM ↑	rFID ↓	PSNR ↑	LPIPS ↓	SSIM ↑
256 × 256												
SD3.5-VAE [7]	84M	*f8d16*	✗	✗	0.19	31.29	0.060	0.88	1.67	31.17	0.056	0.89
FLUX-VAE [13]	84M	*f8d16*	✗	✗	**0.18**	32.80	0.044	0.91	**1.35**	32.74	0.042	0.92
TransVAE-VF-T (Ours)	60M	*f8d16*	✓	✓	0.86	31.73	0.055	0.88	3.12	31.79	0.050	0.91
TransVAE-T (Ours)	60M	*f8d16*	✗	✗	0.34	32.83	0.045	0.91	2.08	32.76	0.043	0.92
TransVAE-VF-L (Ours)	719M	*f8d16*	✓	✓	0.68	32.89	0.042	0.92	2.68	32.85	0.039	0.92
TransVAE-L (Ours)	719M	*f8d16*	✗	✗	0.28	**33.30**	**0.041**	**0.92**	1.84	**33.27**	**0.038**	**0.93**
512 × 512												
SD3.5-VAE [7]	84M	*f8d16*	✗	✗	0.11	33.54	0.060	0.91	0.93	32.56	0.063	0.90
FLUX-VAE [13]	84M	*f8d16*	✗	✗	0.05	35.40	0.042	0.94	**0.58**	34.30	0.045	0.93
TransVAE-VF-T (Ours)	60M	*f8d16*	✓	✓	0.08	33.88	0.056	0.91	1.25	33.01	0.059	0.91
TransVAE-T (Ours)	60M	*f8d16*	✗	✗	0.06	35.61	0.043	0.94	0.75	34.46	0.045	0.93
TransVAE-VF-L (Ours)	719M	*f8d16*	✓	✓	0.09	34.54	0.053	0.93	0.85	33.85	0.053	0.93
TransVAE-L (Ours)	719M	*f8d16*	✗	✗	**0.05**	**35.89**	**0.040**	**0.94**	0.64	**34.68**	**0.041**	**0.94**
1024 × 1024												
SD3.5-VAE [7]	84M	*f8d16*	✗	✗	0.03	37.61	0.034	0.97	0.31	36.80	0.038	0.97
FLUX-VAE [13]	84M	*f8d16*	✗	✗	0.01	39.84	0.023	0.98	**0.16**	39.13	0.025	0.98
TransVAE-VF-T (Ours)	60M	*f8d16*	✓	✓	0.02	38.66	0.035	0.96	0.33	37.94	0.044	0.96
TransVAE-T (Ours)	60M	*f8d16*	✗	✗	0.02	40.15	0.024	0.98	0.21	39.48	0.025	0.98
TransVAE-VF-L (Ours)	719M	*f8d16*	✓	✓	0.02	39.65	0.030	0.98	0.28	38.90	0.045	0.97
TransVAE-L (Ours)	719M	*f8d16*	✗	✗	**0.01**	**40.67**	**0.023**	**0.98**	0.17	**40.24**	**0.022**	**0.98**

considering both reconstruction and alignment, we train all scaling variants of different model architectures for 3 epochs to analyse the early convergence curve. The TransVAE scaling configurations are in Tab. 2, and others are demonstrated in the Supplementary. We also explore the effectiveness of Swin Transformer [15] in the VAE architecture, as it can also combine the local and global representation.

We plot the validation loss curve of reconstruction and alignment in Fig. 9(a) and the visualization results of each model with the corresponding PSNR value in Fig. 9(b). The loss curve and PSNR are averaged over three random seeds, and we can draw the following conclusions. (1) As evidenced by the loss curve, our proposed TransVAE produces clear scaling benefits on both reconstruction and representation alignment. To be specific, the validation losses of the two tasks are obviously better when scaling up. (2) The ViT-VAE requires a substantial amount of parameters to reconstruct the details, and would suffer from high-frequency artifacts as the model size is insufficient to handle the detailed pixel-level representation. (3) Though both validation losses of Swin-VAE converge faster with model scaling, its subtle visual details still underperform and result in low PSNR during inference. (4) The CNN-VAE and our TransVAE are the two models that successfully achieve scalability. Though CNN-VAE has marginally better reconstruction loss than ours, its PSNR still behaves worse, and the reconstruction gain brought by scaling the size of CNN-VAE is limited, as shown by the loss curve and the visualization. Note that it is less meaningful to compare the value of alignment loss across different architectures of models, because the VF loss can be simply controlled or loosened by setting a different margin.

To further explore the scaling behavior with more training epochs, we run a 50-epoch training with the largest 2.3B TransVAE-G variant. As shown by the loss curve in Fig. 10 and Fig. 11, 2.3B TransVAE still demonstrates a consistent scaling benefit in both tasks. In addition, we measure the linear probing accuracy



(a) Reconstruction and alignment loss curves of different models at scale

(b) Reconstruction Results of Different Models when Scaling

Fig. 9: Comparison of CNN-VAE, ViT-VAE, Swin-VAE, and our proposed TransVAE at scale. (a) We show the validation loss on reconstruction and alignment for models. (b) Visualization results of models trained for 3 epochs, demonstrating the learning efficiency of TransVAE over baselines.

Table 5: Linear Probing Accuracy of $f16d32$ VAEs at 50 epochs.

Variant	Acc. (%)
TransVAE-T	10.91
VA-VAE-VF	42.05
TransVAE-VF-T	44.13
TransVAE-VF-L	53.77
TransVAE-VF-G	**65.18**

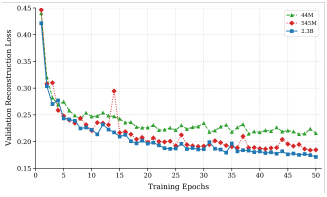


Fig. 10: Reconstruction loss over 50 epochs.

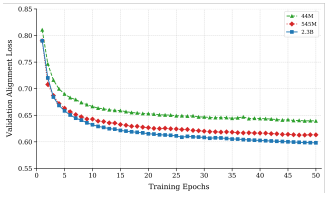


Fig. 11: Alignment loss over 50 epochs.

in the latent space of different VAEs trained with alignment for 50 epochs, as shown in Tab. 5. Our TransVAE has a better latent space with more discriminative semantic representation, which will facilitate downstream generation tasks.

In conclusion, our TransVAE can truly scale up and bring in profound performances both in the basic reconstruction task of VAE and the representation alignment task to be more downstream-friendly. More discussion and evidence are shown in the Supplementary Materials.

4.4 Downstream Tasks Performance

We evaluate the latent space of pretrained VAEs by image generation with LightningDiT. We first train a LightningDiT-B/2 with $f8d16$ VAEs, as shown in

Table 6: FID-50k on ImageNet 256 × 256.

Method	Tokenizer	#params	Epochs	rFID	gFID (w/o CFG)	gFID (w/ CFG)
DiT [20]	SD-VAE [21]	675M	1400	0.61	9.62	2.27
SiT [19]		675M	1400		8.61	2.06
FasterDiT [30]		675M	400		7.91	2.03
MDT [8]		675M	1300		6.23	1.79
REPA [33]		675M	800		5.90	1.42
LightningDiT-XL [31]	VA-VAE-VF [31]	675M	64	0.28	5.14	2.11
		675M	80		4.51	1.96
LightningDiT-XL [31]	TransVAE-VF-T	675M	64	0.41	4.72	2.04
		675M	80		3.81	1.90
LightningDiT-XL [31]	TransVAE-VF-L	675M	64	0.37	4.38	1.96
		675M	80		3.58	1.85

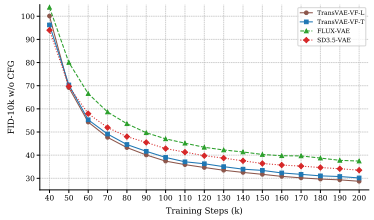


Fig. 12: FID-10k w/o CFG on LightningDiT-B with $f8d16$ VAE.

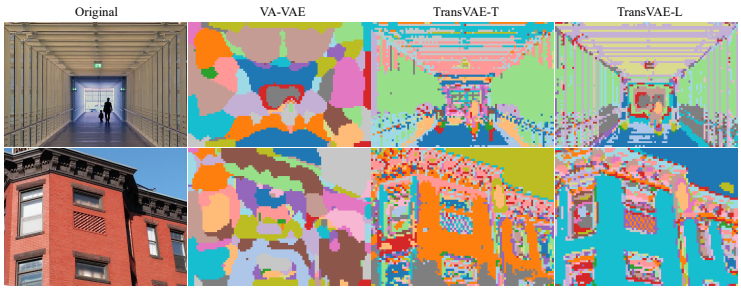


Fig. 13: Zero-shot segmentation visualization on ADE20K using frozen encoder features from stages 2, 3, and 4. Pixel-level K-Means clusters show that deeper TransVAE features yield more coherent object-level regions.

Fig. 12. Trained with alignment loss, our TransVAE-VF-L achieves a similar FID at 100k steps as FLUX-VAE's at 200k steps, exhibits a 2× faster convergence. We further train LightningDiT-XL/1 with $f16d32$ VAEs, as shown in Tab. 6. Our TransVAE-VF achieves better FID-50k performance than VA-VAE-VF when training 64 and 80 epochs, either with or without CFG, while also demonstrating scaling benefit in generation. Both experiments support the scaling properties of TransVAE-VF in the downstream tasks. Thus, scaling TransVAE demonstrates a better trade-off between reconstruction and downstream-friendly latent space.

Qualitative zero-shot segmentation analysis. We further visualize frozen encoder features via zero-shot segmentation on ADE20K. For each image, we extract stage-wise encoder features, upsample them to a common resolution, and apply K-Means clustering to pixel-level features without training any segmentation head. As shown in Fig. 13, deeper TransVAE features form more coherent object-level regions and cleaner boundaries than the baseline VAE, suggesting stronger semantic structure in the learned representation.

4.5 Inference Latency Comparison

We measure the inference speed of different VAEs under the same setting. The inference speed on 256 × 256 resolution is measured using one H20 GPU with

Table 7: Inference Latency Analysis. Throughput represents the number of images to be encoded and decoded per second. All the measurements are conducted by a single H20 GPU on 256×256 resolution with a batch size of 16 under BFloat16 precision.

Model	Variant	#Params (Total)	Encode Throughput ↑	Decode Throughput ↑	Total GPU Mem ↓
FLUX-VAE	$f8d16$	84M	235	118	4.6G
TransVAE-T	$f8d16$	60M	119	103	3.1G
TransVAE-L	$f8d16$	719M	35	31	6.9G
VA-VAE	$f16d32$	70M	371	220	3.8G
TransVAE-T	$f16d32$	44M	193	168	3.7G
TransVAE-L	$f16d32$	545M	79	70	6.8G
DC-AE	$f32d32$	323M	169	132	1.8G
TransVAE-B	$f32d32$	287M	111	95	3.8G

a batch size of 16 and BFloat16 precision. As shown in Tab. 7, our current implementation prioritizes architectural clarity over kernel-level optimization. We will optimize the end-to-end pipeline in future work. The only optimized module is the Flash Attention, which substantially reduces the GPU memory cost, while the throughput of the whole model is not fully optimized. We will optimize the whole framework for practical implementations in the future.

5 Conclusion

In this paper, we systematically study the architectural components of ViT-based VAEs and propose **TransVAE**, a hybrid CNN–ViT VAE built with minimal yet principled modifications. By unifying a CNN stem for efficient local detail modeling with a Transformer backbone for global context, TransVAE addresses three practical limitations of prior ViT-VAEs: slow convergence, weak resolution extrapolation, and inconsistent scaling gains. Extensive experiments show that TransVAE improves learning efficiency, supports robust single-resolution training with higher-resolution inference, and exhibits predictable scaling from 44M to 2.3B parameters. Overall, TransVAE provides a simple and scalable tokenizer architecture with a favorable trade-off between reconstruction fidelity and generation-friendly latents, leaving room for future improvements in alignment objectives and optimized implementations.

Acknowledgements

This work is supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (JYB2025XDXM103) and the National Nature Science Foundation of China (62425114, 62121002, U23B2028, 62272436). This work is also supported in part by GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC, and in part by the advanced computing resources provided by the Supercomputing Center of the USTC.

References

1. 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: Forty-second International Conference on Machine Learning (2025)
2. Chen, H., Wang, Z., Li, X., Sun, X., Chen, F., Liu, J., Wang, J., Raj, B., Liu, Z., Barsoum, E.: Softvq-vae: Efficient 1-dimensional continuous tokenizer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 28358–28370 (2025)
3. Chen, J., Cai, H., Chen, J., Xie, E., Yang, S., Tang, H., Li, M., Han, S.: Deep compression autoencoder for efficient high-resolution diffusion models. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=wH8XXUOUZU
4. Chen, J., Cai, H., Chen, J., Xie, E., Yang, S., Tang, H., Li, M., Lu, Y., Han, S.: Deep compression autoencoder for efficient high-resolution diffusion models. arXiv preprint arXiv:2410.10733 (2024)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
7. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first International Conference on Machine Learning (2024)
8. Gao, S., Zhou, P., Cheng, M.M., Yan, S.: Masked diffusion transformer is a strong image synthesizer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23164–23173 (2023)
9. Hansen-Estruch, P., Yan, D., Chung, C.Y., Zohar, O., Wang, J., Hou, T., Xu, T., Vishwanath, S., Vajda, P., Chen, X.: Learnings from scaling visual tokenizers for reconstruction and generation. arXiv preprint arXiv:2501.09755 (2025)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. Kingma, D.P.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)
13. Labs, B.F.: Frontier ai lab. https://blackforestlabs.ai/, accessed: 2024-11-12
14. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
15. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)

16. Lu, Z., Liu, C., Chang, X., Zhang, Y., Xie, H.: Dhvt: Dynamic hybrid vision transformer for small dataset recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
17. Lu, Z., Xie, H., Liu, C., Zhang, Y.: Bridging the gap between vision transformers and convolutional neural networks on small datasets. Advances in Neural Information Processing Systems **35**, 14663–14677 (2022)
18. Ma, C., Jiang, Y., Wu, J., Yang, J., Yu, X., Yuan, Z., Peng, B., Qi, X.: Unitok: A unified tokenizer for visual generation and understanding. arXiv preprint arXiv:2502.20321 (2025)
19. 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. arXiv preprint arXiv:2401.08740 (2024)
20. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4195–4205 (2023)
21. 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. pp. 10684–10695 (2022)
22. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
23. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
24. Teng, H., Jia, H., Sun, L., Li, L., Li, M., Tang, M., Han, S., Zhang, T., Zhang, W., Luo, W., et al.: Magi-1: Autoregressive video generation at scale. arXiv preprint arXiv:2505.13211 (2025)
25. Vahdat, A., Kautz, J.: Nvae: A deep hierarchical variational autoencoder. Advances in neural information processing systems **33**, 19667–19679 (2020)
26. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. Advances in neural information processing systems **30** (2017)
27. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 568–578 (2021)
28. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22–31 (2021)
29. Yang, J., Li, T., Fan, L., Tian, Y., Wang, Y.: Latent denoising makes good visual tokenizers. arXiv preprint arXiv:2507.15856 (2025)
30. Yao, J., Cheng, W., Liu, W., Wang, X.: Fasterdit: Towards faster diffusion transformers training without architecture modification. arXiv preprint arXiv:2410.10356 (2024)
31. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 15703–15712 (2025)
32. Yu, Q., Weber, M., Deng, X., Shen, X., Cremers, D., Chen, L.C.: An image is worth 32 tokens for reconstruction and generation. Advances in Neural Information Processing Systems **37**, 128940–128966 (2024)
33. 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. arXiv preprint arXiv:2410.06940 (2024)

34. Yuan, K., Guo, S., Liu, Z., Zhou, A., Yu, F., Wu, W.: Incorporating convolution designs into visual transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 579–588 (October 2021)
35. Zhang, B., Sennrich, R.: Root mean square layer normalization. Advances in neural information processing systems **32** (2019)
36. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)