MIMFlow: Integrating Masked Image Modeling with Normalizing Flows for End-to-End Image Generation

Yang Chen[1,2], Xiaowei Xu[2], Shuai Wang[1], Xinwen Zhang[1,2], Qiushi Guo[2], Tiezheng Ge[2], and Limin Wang[1,3,⋆]

[1] State Key Laboratory for Novel Software Technology, Nanjing University
[2] Alibaba Group [3] Shanghai AI Lab
⋆ Corresponding author: lmwang@nju.edu.cn
Code: `https://github.com/MCG-NJU/MIMFlow`

Abstract. Normalizing Flows (NFs) are powerful generative models capable of exact density estimation and sampling. However, their strict invertibility often forces the model to exhaust its capacity on low-level pixel details, hindering the capture of high-level semantic structures. While Masked Image Modeling (MIM) has excelled in representation learning, its integration into generative pipelines has remained largely modular and disjointed. In this paper, we propose MIMFlow, a unified end-to-end framework that jointly optimizes latent semantics, pixel reconstruction, and generative flow. By employing a VAE encoder to infer semantic latent from masked images, MIMFlow achieves a principled decoupling of the generative task: the Normalizing Flow focuses on modeling a simplified, low-frequency semantic manifold, while a specialized decoder handles high-frequency synthesis. This design effectively resolves the inherent capacity bottleneck of NFs, allowing the model to prioritize global structural coherence over redundant noise. Empirical results on ImageNet 256×256 show that MIMFlow-L reaches 71.3% linear probing accuracy and an FID of 2.50. Despite using only 128 tokens (50% fewer than standard models), it yields a 32.8% performance gain over similar-scale NF baselines.

Keywords: Image Generation · Normalizing Flow · MIM

1 Introduction

Normalizing Flows (NFs) are generative models that map a complex data distribution to a simple prior distribution through a series of invertible and differentiable transformations [4,8,15,16,23,27,39,42,43,52,53,55]. A primary advantage of NFs is their ability to perform both exact density estimation and sampling within a single network. Besides, from the perspective of flow matching, NFs can be viewed as the expansion of an Ordinary Differential Equation (ODE), allowing for end-to-end optimization of the entire flow process [2]. Recent work, such as SimFlow [53], has further demonstrated that NFs can provide highly

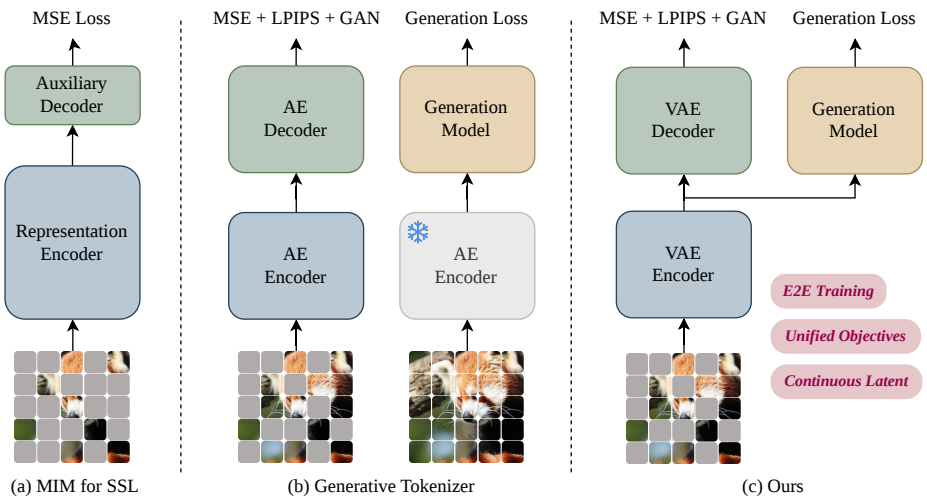


Fig. 1: MIM in Different Paradigms. (a) Self-Supervised Learning: Employs high-ratio masking as a self-supervised proxy task for representation learning. (b) Generative Tokenizers: A two-stage approach where the latent space is pre-trained with MIM before training a separate generative model. (c) MIMFlow (Ours): A unified framework that jointly optimizes latent semantics, pixel reconstruction, and generative flow in an end-to-end manner.

expressive priors for Variational Autoencoders (VAEs) by leveraging their likelihood estimation capabilities. However, strict invertibility forces NFs to prioritize low-level pixel details over high-level semantics, exhausting model capacity and hindering generative quality.

While NFs struggle to capture global structures, Masked Image Modeling (MIM, Fig. 1a) has established itself as a cornerstone of self-supervised representation learning [18, 46]. Despite its success in discriminative tasks, the role of MIM in generative modeling remains relatively under-explored. Recent literature suggests that integrating discriminative features can significantly bolster generative performance, with some approaches distilling knowledge from pre-trained representation models to guide the generation process [26, 51, 57]. Furthermore, RAE [40, 54] directly employs representation models as VAE encoders. Specifically, they found that MAE lags behind DINO in these generative frameworks. This likely stems from the mismatch between MAE's high-mask reconstruction and the continuous distribution modeling required for synthesis. **Thus, whether semantic-focused MIM can effectively enhance generative models remains an open question.**

Recent attempts to incorporate MIM into generative pipelines have primarily focused on refining visual tokenizers. Specifically, works such as MAETok [1] and DeTok [48] have pioneered the use of masking as a denoising or discriminative objective to enhance the robustness of latent structures. Building on this, VTP [49] demonstrated that unifying MIM with contrastive learning can further foster a

high-level understanding essential for tokenizer scalability. However, these methods largely treat the tokenizer as a modular, standalone precursor, separating representation learning from the core generative process (Fig. 1b). In contrast, our method jointly optimizes representation, reconstruction and generation in an end-to-end framework (Fig. 1c). This design enables a principled decoupling of the generative task: the NF is dedicated to modeling the low-frequency semantic manifold, while a specialized decoder handles high-frequency synthesis. By reducing the burden on NFs to capture pixel-level noise, MIMFlow mitigates a central capacity bottleneck of traditional NFs and offers an NF-oriented recipe for more semantic latent modeling.

Specifically, our MIMFlow employs a VAE encoder with learnable tokens to extract stable latent representations from masked images. These latents are then optimized through a dual-objective framework: a NF performs exact density estimation, while a decoder focuses on pixel-level reconstruction. This architecture facilitates a strategic division of labor: by mandating the encoder to infer missing spatial context, the resulting latent space is biased toward global structural coherence rather than redundant local noise. Consequently, the NF can model a simpler semantic manifold instead of dense pixel-level correlations. Crucially, linear probing evaluations reveal a substantial increase in classification accuracy within the latent space, validating the enhanced semantic quality of our representations. Finally, on the ImageNet 256×256 benchmark, MIMFlow improves similar-scale NF baselines while using fewer latent tokens.

In summary, our main contributions are as follows:

- **MIMFlow Framework:** We propose an end-to-end framework that unifies representation and generation, moving beyond the modular limitations of existing tokenizer-based methods.
- **Principled Decoupling:** We introduce a strategy to decouple semantic modeling from pixel-level synthesis, effectively resolving the inherent capacity bottleneck of NFs.
- **Empirical Excellence:** We demonstrate that MIMFlow significantly enhances latent semantics and achieves superior generative performance on the ImageNet 256 × 256 benchmark.

2 Related Work

Normalizing Flows [7–10, 13, 17, 21, 23, 24, 31, 36] provide a mathematically principled framework for bidirectional mapping between data and latent spaces. Early works like RealNVP [8] and Glow [23] established exact log-likelihood estimation through coupling layers, though they struggled to scale to high-resolution synthesis. Recent advancements [4, 14, 42, 52] have revitalized the field by integrating autoregressive Transformers and latent-space modeling, effectively leveraging NFs for both high-fidelity generation and semantic alignment. Building on this, SimFlow [53] demonstrates that NFs can serve as highly expressive probability estimators to replace restrictive VAE priors, significantly enhancing generative performance. While these models focus on architectural scaling or

post-hoc alignment, our work is the first to integrate MIM objectives directly into the end-to-end training of NFs. By unifying self-supervised representation learning with generative flow optimization, we fully exploit the dual potential of NFs for simultaneous robust feature extraction and high-quality synthesis.

Masked Image Modeling has emerged as a dominant paradigm in self-supervised learning, with methods like MAE [18] and SimMIM [46] demonstrating that reconstructing masked inputs forces encoders to learn robust structural features. While initially designed for discriminative tasks, it has also been utilized in generative frameworks. Specifically, recent literature explores distilling knowledge from pre-trained vision foundation models to guide diffusion processes [51, 57]. Within the context of visual tokenizers, MAETok [1] and DeTok [48] employ masking as a denoising or discriminative auxiliary task to enhance latent robustness. Unlike methods above, MIMFlow integrates MIM directly into end-to-end generative training, ensuring that the generative flow is conditioned on highly compressed, semantic-rich features.

End-to-End Joint Training. Generative models typically follow a two-stage pipeline—training a VAE for reconstruction and then a generative model on the frozen latent space. However, this can lead to a mismatch where high reconstruction fidelity fails to translate into generative quality. To bridge this gap, REPA-E [26] and SimFlow [53] have pioneered end-to-end joint training, with SimFlow optimizing NFs and VAEs simultaneously from scratch. While our MIMFlow adopts this end-to-end philosophy, it distinguishes itself by incorporating a masked bottleneck to explicitly decouple semantic modeling from texture synthesis. This approach shields the NF from high-frequency noise and ensures the learned latent space is inherently optimized for global structure.

3 Method

In this section, we present the architecture and mathematical formulation of **MIMFlow**, a unified generative framework that integrates MIM with Normalizing Flows. As illustrated in Fig. 2, the framework comprises three core modules. First, a **Masked Encoder** (E_ϕ) extracts robust latent representations by capturing global structural semantics from masked images. Second, a **Latent Normalizing Flow** (f_θ) performs probabilistic modeling by mapping these representations to a Gaussian prior, enabling exact density estimation as the prior for VAE modeling. Finally, a **Generative Decoder** (D_ψ) handles pixel-level reconstruction and texture synthesis. This hierarchical design ensures that the flow model focuses on global structure while the decoder handles local details. In the following, we detail the latent extraction mechanism, the probabilistic modeling framework, and our multi-stage optimization strategy involving auxiliary supervision and adversarial refinement.

3.1 Semantic Latent Extraction via Learnable Tokens

A fundamental challenge in integrating MIM with NFs lies in the construction of a stable and expressive latent space. Conventional MIM backbones, such as

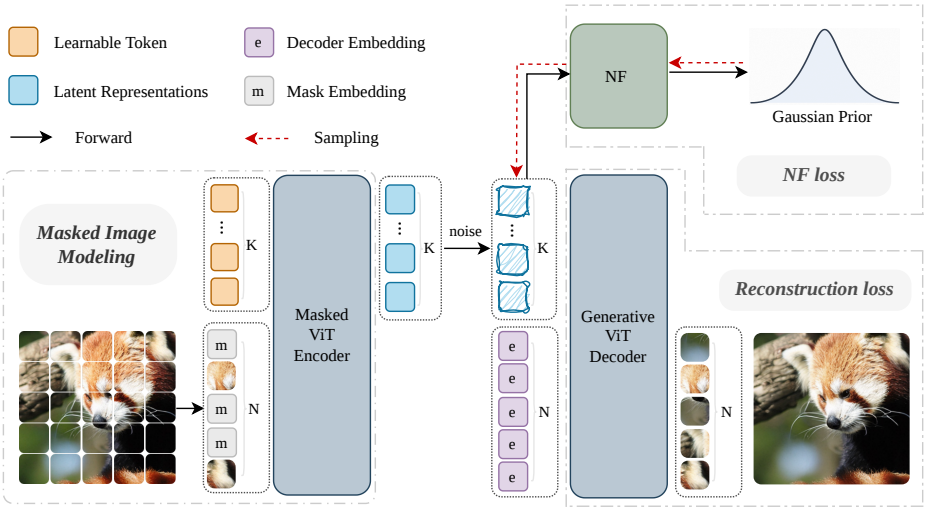


Fig. 2: Structure of MIMFlow. N is the number of image patches, K is the number of learnable latent query tokens, $\mathbf{m}$ is the binary mask, and e denotes learnable decoder embeddings.

MAE [18] and SimMIM [46], present inherent difficulties for density estimation. MAE only processes visible patches, resulting in a latent sequence whose length and positional context vary with the random mask pattern, which imposes an intractable burden on the NF to learn a consistent distribution. Conversely, while SimMIM maintains a fixed sequence length by utilizing mask tokens, the information density within each token fluctuates significantly depending on its masking status. Such stochasticity prevents the NF from converging on a stable semantic manifold.

To resolve these issues, we introduce a **Learnable Token Bottleneck** to extract fixed-dimensional latent representations. We posit that since masked images contain lower information density than full images, they can be more efficiently compressed into a compact latent space. Formally, given an input image $\mathbf{x}$, we partition it into N patches and apply a random masking ratio. The resulting N tokens (including mask tokens) are concatenated with K learnable query tokens, where $K < N$. These $N + K$ tokens are processed by a bidirectional Transformer encoder E_ϕ, enabling the learnable queries to aggregate global semantic information from the partially observed image through self-attention.

After the encoding stage, we extract only the K refined query tokens as our latent representation $\mathbf{z} \in \mathbb{R}^{K \times D}$. Following the practice of SimFlow [53], we inject additive Gaussian noise with a fixed variance to $\mathbf{z}$ during training to facilitate continuous density modeling:

$$\hat{\mathbf{z}} = \mathbf{z} + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{1}$$

where σ is a hyperparameter. This $\hat{\mathbf{z}}$ serves a dual purpose within our framework:

- **Generative Modeling:** It is treated as a 1D sequence and fed into the NF f_θ for exact likelihood estimation, and reverse sampling.
- **Reconstruction:** It is passed to the decoder D_ψ, where it is combined with learnable image embeddings e to reconstruct the original pixel-level signal through cross-modal attention.

This design provides several strategic advantages. First, the fixed-length K tokens provide a stable target for the NF, regardless of the random mask positions. Second, the $K < N$ bottleneck forces the model to discard local pixel redundancies and concentrate on high-level structural semantics. Finally, the flexible masking ratio during training enhances the robustness of the latent space, ensuring that the NF models a manifold that is truly representative of the underlying global data structure.

3.2 Probabilistic Modeling and Joint Optimization

In this section, we formulate MIMFlow as a conditional generative model grounded in the Variational Inference (VI) framework. Specifically, we treat the encoder E_ϕ and decoder D_ψ as a Variational Autoencoder (VAE) where the standard Gaussian prior is replaced by a high-capacity Normalizing Flow f_θ. Unlike standard VAEs that take the complete image as input, our encoder is conditioned on the masked image $\tilde{\mathbf{x}} = \mathbf{x} \odot (1 - \mathbf{m})$, where $\mathbf{m}$ denotes the binary mask.

Variational Inference Perspective. Our objective is to maximize the log-likelihood of the data distribution $p(\mathbf{x}|\mathbf{m})$. Following the variational principle, we introduce an approximate posterior $q_\phi(\mathbf{z}|\tilde{\mathbf{x}})$ and derive the Evidence Lower Bound (ELBO):

$$\log p(\mathbf{x}|\mathbf{m}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\tilde{\mathbf{x}})}[\log p_\psi(\mathbf{x}|\mathbf{z})] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\tilde{\mathbf{x}}) \parallel p_\theta(\mathbf{z})) = \mathcal{L}_{\mathrm{ELBO}}, \quad (2)$$

where $p_\psi(\mathbf{x}|\mathbf{z})$ is the reconstruction likelihood and $p_\theta(\mathbf{z})$ is the prior distribution modeled by the Normalizing Flow.

In our framework, we define $q_\phi(\mathbf{z}|\tilde{\mathbf{x}})$ as a Gaussian distribution with fixed variance σ^2 centered at the encoder's output $E_\phi(\tilde{\mathbf{x}})$. This simplifies the KL divergence term to the cross-entropy between the posterior and the flow-based prior (omitting constant entropy terms):

$$D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\tilde{\mathbf{x}}) \parallel p_\theta(\mathbf{z})) \propto -\mathbb{E}_{q_\phi(\mathbf{z}|\tilde{\mathbf{x}})}[\log p_\theta(\mathbf{z})] + C. \quad (3)$$

Density Estimation via Normalizing Flow. The prior $p_\theta(\mathbf{z})$ is parameterized by an invertible transformation f_θ that maps the complex latent $\mathbf{z}$ to a simple base distribution $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. Using the change-of-variables formula, the exact log-likelihood of the latent $\mathbf{z}$ can be computed as:

$$\log p_\theta(\mathbf{z}) = \log p_\epsilon(f_\theta(\mathbf{z})) + \log \left| \det \frac{\partial f_\theta(\mathbf{z})}{\partial \mathbf{z}} \right|, \quad (4)$$

where the first term represents the density under the Gaussian prior, and the second term is the log-determinant of the Jacobian, accounting for the volume change induced by the transformation. By optimizing this term, the NF effectively learns to warp the simple Gaussian into a sophisticated semantic manifold that captures the global dependencies of the image.

Joint Training Objective. Combining the reconstruction and the flow-based prior, we arrive at the total loss function for MIMFlow. The reconstruction term $\mathbb{E}_{q_\phi}[\log p_\psi(\mathbf{x}|\mathbf{z})]$ is implemented as a combination of ℓ_2 loss and perceptual loss to ensure both pixel-level fidelity and semantic coherence:

$$\mathcal{L}_{\text{rec}} = \|\mathbf{x} - D_\psi(\mathbf{z})\|_2^2 + \lambda_{\text{perc}}\mathcal{L}_{\text{LPIPS}}(\mathbf{x}, D_\psi(\mathbf{z})), \tag{5}$$

where $\mathbf{z} \sim q_\phi(\mathbf{z}|\tilde{\mathbf{x}})$. Notably, since $\mathbf{z}$ is extracted from the masked image $\tilde{\mathbf{x}}$, the decoder is forced to perform both *denoising* and *inpainting*, which encourages the latent space to prioritize high-level structure over local noise.

The final joint loss is defined as:

$$\mathcal{L}_{\text{prob}} = \mathcal{L}_{\text{rec}} + \beta\mathcal{L}_{\text{NF}}, \tag{6}$$

where $\mathcal{L}_{\text{NF}} = -\log p_\theta(\mathbf{z})$ is the negative log-likelihood (NLL) provided by the NF. By jointly optimizing ϕ, ψ and θ, MIMFlow ensures that the latent space is simultaneously optimized for representation, reconstruction and density estimation, leading to a more robust and expressive generative model.

3.3 Auxiliary Supervision and Adversarial Refinement

To further enrich the latent representations and enhance the perceptual quality of the synthesized images, we incorporate auxiliary semantic supervision and a subsequent adversarial fine-tuning stage.

Auxiliary Feature Prediction. Following the design of generative tokenizers such as MAETok [1], we augment the training objective with an auxiliary discriminative task. In addition to the primary pixel decoder, a lightweight MLP-based auxiliary decoder D_{aux} is employed to predict high-level features $\mathbf{F}_{target}$ (e.g., from DINO [30] or CLIP [34]) directly from the latent $\mathbf{z}$. The auxiliary loss is defined as:

$$\mathcal{L}_{\text{aux}} = \|D_{aux}(\mathbf{z}) - \text{sg}(\mathbf{F}_{target}(\mathbf{x}))\|_2^2. \tag{7}$$

By supervising the latent space with pre-trained discriminative priors, this objective encourages the encoder to capture more robust semantic context beyond low-level pixel statistics. During the end-to-end training phase, the total loss is formulated as $\mathcal{L} = \mathcal{L}_{\text{rec}} + \beta\mathcal{L}_{\text{NF}} + \gamma\mathcal{L}_{\text{aux}}$, ensuring that the Normalizing Flow models a latent distribution that is both reconstructive and semantically rich.

Adversarial Fine-tuning. While the joint training phase establishes a structured latent space, the resulting reconstructions often lack the high-frequency details necessary for photorealistic generation. To address this, we perform a targeted fine-tuning of the pixel decoder D_ψ. In this stage, we introduce a patch-based discriminator $\mathcal{D}$ and optimize the model using a combination of reconstruction loss and GAN loss:

$$\mathcal{L}_{\text{FT}} = \mathcal{L}_{\text{rec}} + \alpha \mathcal{L}_{\text{GAN}}(D_\psi, \mathcal{D}). \tag{8}$$

Crucially, during fine-tuning, the encoder E_ϕ continues to receive the *masked* image $\tilde{\mathbf{x}}$ as input. This preserves **distributional consistency** in the latent space: the decoder continues to observe latents from the same masked posterior family that the NF was trained to model, while sampling draws latents from the corresponding NF-modeled manifold rather than relying on an explicit mask. This allows the decoder to focus on synthesizing sharp textures without disrupting the probabilistic alignment between the generative flow and the latent representations.

4 Experiment

4.1 Experimental Setup

Datasets and Evaluation Metrics We conduct our experiments on the ImageNet dataset at a 256×256 resolution [5]. To comprehensively evaluate the generative performance, we report the Fréchet Inception Distance (FID) [19], Inception Score (IS) [38], Precision, and Recall [25], all calculated using the evaluation suite provided by ADM [6]. For reconstruction quality, we compute the reconstruction FID (rFID) on the ImageNet validation set. Furthermore, to assess the semantic quality of the learned latent space, we perform linear probing on the validation set following REPA's [51] protocol, reporting the top-1 classification accuracy.

Implementation Details Our framework consists of two primary components: the Normalizing Flow prior and the Transformer-based Autoencoder.

Normalizing Flow Architecture We adopt the STARFlow [14] architecture for density estimation in the latent space. Our model, designated as **STARFlow-L**, comprises approximately 482M parameters with a hidden dimension of 1024. The backbone consists of seven 2-layer blocks followed by a final 20-layer block.

Table 1: Baseline Construction (last line), evaluated with 10K samples.

	w/o CFG	w/ CFG	
Configuration	gFID ↓	gFID ↓	sFID ↓
STARFlow-L	50.22	7.79	20.54
− Softplus	48.24	7.65	18.98
+ Gated Attn.	47.44	7.45	18.64
+ End-to-End	**11.24**	**5.99**	–

Fig. 3: Selected Samples on ImageNet 256 × 256 from MIMFlow-L. We use classifier-free guidance equal to 2.0.

We iteratively construct our baseline starting from a STARFlow-L model trained on a fixed VAE latent space. The refinement process involves: (1) removing the **softplus** operation on the scaling factors to improve numerical flexibility; (2) incorporating **gated attention** [33] mechanisms to enhance feature interaction; and (3) transitioning to full **end-to-end (e2e)** training with GAN loss, and without auxiliary loss. Relative to SimFlow-L, this baseline changes only the NF backbone, which makes the close FID (3.70 vs. 3.72) an apples-to-apples check. The performance gains from each stage are summarized in Tab. 1. For optimization, we use the AdamW optimizer with a learning rate of 1×10^{-4} and a global batch size of 256. The final model is trained end-to-end for 90 epochs, followed by 2 additional epochs of decoder fine-tuning.

Transformer-based Autoencoder To accommodate the requirements of MIM, we replace the traditional convolutional stages of the SD-VAE[3] [37] with Transformer-based MAETok [1]. Notably, we train the entire framework from scratch without loading any pre-trained parameters for the MAETok backbone. Our latent space is configured with 128 tokens, each having a dimensionality of 64, and is integrated with 1D positional encodings. During training, we adopt a random masking strategy to facilitate the MIM objective, with a default mask ratio ranging from 0.4 to 0.6. As shown in Tab. 2, while the ViT-B backbone increases the parameter count, it significantly reduces the computational overhead in terms of FLOPs compared to the convolutional VAE, making it more suitable for high-resolution processing

Table 2: Comparison of Architectural Complexity

Model	Parameters	FLOPs
SD-VAE	83.7M	446.21G
MAETok	172.0M	71.25G

Table 3: System performance comparison on ImageNet 256 × 256 class-conditioned generation. Gray rows denote larger-scale models within the latent normalizing flow category, provided for reference.

Method	#Tokens	#Params	W/o guidance				W/ guidance			
			gFID↓	IS↑	Prec.↑	Rec.↑	gFID↓	IS↑	Prec.↑	Rec.↑
Pixel Space										
ADM [6]	-	554M	10.94	101.0	0.69	0.63	3.94	215.8	0.83	0.53
RIN [22]	-	410M	3.42	182.0	-	-	-	-	-	-
PixelFlow [3]	4096	677M	-	-	-	-	1.98	282.1	0.81	0.60
PixNerd [44]	1024	700M	-	-	-	-	2.15	297.0	0.79	0.59
SiD2 [20]	-	-	-	-	-	-	1.38	-	-	-
TARFlow [52]	1024	1.4B	-	-	-	-	4.69	-	-	-
JetFormer [42]	256	2.8B	-	-	-	-	6.64	-	0.69	0.56
FARMER [55]	1024	1.9B	-	-	-	-	3.60	269.2	0.81	0.51
Latent Autoregressive										
VAR [41]	680	2.0B	1.92	323.1	0.82	0.59	1.73	350.2	0.82	0.60
MAR [28]	256	943M	2.35	227.8	0.79	0.62	1.55	303.7	0.81	0.62
xAR [35]	-	1.1B	-	-	-	-	1.24	301.6	0.83	0.64
Latent Diffusion										
DiT [32]	256	675M	9.62	121.5	0.67	0.67	2.27	278.2	0.83	0.57
MaskDiT [56]	-	675M	5.69	177.9	0.74	0.60	2.28	276.6	0.80	0.61
SiT [29]	256	675M	8.61	131.7	0.68	0.67	2.06	270.3	0.82	0.59
MDTv2 [11]	256	675M	-	-	-	-	1.58	314.7	0.79	0.65
REPA [51]	256	675M	5.78	158.3	0.70	0.68	1.29	306.3	0.79	0.64
VA-VAE [50]	256	675M	2.17	205.6	0.77	0.65	1.35	295.3	0.79	0.65
DDT [45]	256	675M	6.27	154.7	0.68	0.69	1.26	310.6	0.79	0.65
REPA-E [26]	256	675M	1.69	219.3	0.77	0.67	1.12	302.9	0.79	0.66
RAE [54]	256	839M	1.51	242.9	0.79	0.63	1.13	262.6	0.78	0.67
Latent Normalizing Flows										
FlowBack-XL [4]	1024	831M	-	-	-	-	4.18	240.8	-	-
STARFlow-XXL [14]	1024	1.4B	-	-	-	-	2.40	-	-	-
FAE-NF-XXL [12]	256	1.4B	-	-	-	-	2.67	-	-	-
SimFlow-XXL [53]	256	1.4B	10.13	124.7	0.71	0.61	1.91	284.4	0.82	0.60
SimFlow-L [53]	256	475M	33.53	-	-	-	3.72	-	-	-
Baseline (Ours)	256	482M	-	-	-	-	3.70	-	-	-
MIMFlow-L (Ours)	128	482M	3.64	158.6	0.78	0.60	2.50	233.5	0.82	0.57

4.2 Main Results

We compare MIMFlow-L with state-of-the-art (SOTA) generative models on the ImageNet 256 × 256 benchmark, including Pixel-space models, Latent Autoregressive (AR) models, Latent Diffusion Models (LDMs), and existing Latent Normalizing Flows. The quantitative results are summarized in Table 3. Besides, qualitative results are shown in Fig. 3, where MIMFlow-L produces high-fidelity images with consistent global structures.

Superiority within the NF Paradigm. MIMFlow-L demonstrates a significant performance leap over existing Normalizing Flow baselines. Compared to the closely related SimFlow-L, which shares a similar parameter count (∼480M), MIMFlow-L improves the FID from 3.72 to **2.50**—a 32.8% reduction. Notably,

[3] `https://huggingface.co/stabilityai/sd-vae-ft-ema`

Table 4: Hardware efficiency comparison on ImageNet 256 × 256 using 8×H800 GPUs. We use no auxiliary supervision in this comparison; the per-GPU batch sizes are 16 for training and 256 for inference.

Model	Tokens	Params	Train Mem.	Train Speed	Sample / img
SimFlow-L	256	475M	52.3GB	2.83 it/s	0.020s
MIMFlow-L	**128**	**482M**	**37.6GB**	**3.11 it/s**	**0.011s**

our model with only 482M parameters outperforms much larger models like FAE-NF-XXL (FID 2.67), which utilizes nearly 3 × the parameters (1.4B) and is built upon the advanced RAE latent space. It also closely approaches the performance of STARFlow-XXL (FID 2.40). This suggests that integrating MIM allows the flow model to learn a more efficient and structured semantic manifold, extracting higher generative value from the same parameter budget.

Efficiency via Token Compression. A key highlight of MIMFlow is its token efficiency. While most latent models (e.g., DiT, LDM, SimFlow) operate on a $16 \times 16 = 256$ token grid, and some even require 1024 tokens (STARFlow, FlowBack), MIMFlow-L achieves strong NF results using only **128 tokens** for sampling.

1. *Reduced Computational Complexity:* By halving the sequence length compared to standard VAE-based models, the computational overhead of the Normalizing Flow (which typically scales quadratically or involves deep Transformer blocks) is significantly reduced.
2. *Information Density:* This 128-token bottleneck supports our hypothesis in Sec. 3.1: because the NF is less exposed to high-frequency noise through MIM, it can represent the global image structure more compactly. This allows MIMFlow to maintain high precision (0.82) while operating at a fraction of the sequence length used by many competitors. Appendix Table **??** further reports the latency benefit of reducing the token count.

We further compare L-scale flow models under the same hardware setting in Tab. 4. Despite using slightly more parameters (482M vs. 475M), MIMFlow-L reduces training memory from 52.3GB to 37.6GB, improves throughput from 2.83 to 3.11 iterations per second, and reduces per-image sampling time from 0.020s to 0.011s. This corresponds to a 28% memory reduction, a 10% throughput improvement, and nearly halved sampling time; the lower memory footprint also makes training feasible on 8×A100 GPUs with a total batch size of 256.

Impact of Guidance and Semantic Quality. As shown in Tab. 3, the gap between "w/o guidance" and "w/ guidance" performance is notably narrower for MIMFlow compared to early flow models. Without guidance, MIMFlow-L achieves an FID of 3.64, which is significantly better than the 10.13 reported for SimFlow-XXL. This indicates that the latent space learned through joint MIM and NF optimization is inherently more structured and semantically coherent, requiring less external guidance to produce high-quality samples.

Table 5: Ablation studies on ImageNet 256×256. **Bold** indicates the default configuration. **Acc.** denotes linear probing accuracy on the encoder's representations *before* the projection layer. **Mix** in Tab. 5a represents a 50/50 probability of using *None* and *0.4–0.6* masking ratios. 10K images are sampled for evaluation.

(a) Masking Strategies

Ratio	rFID↓	gFID↓	IS↑	Prec.↑	Rec.↑	Acc.↑
None	23.5	29.0	65.5	0.59	0.56	56.6
0.2–0.4	8.66	24.47	60.1	0.59	0.57	54.2
0.4–0.6	**3.40**	**12.82**	**88.6**	**0.70**	**0.66**	**71.3**
0.6–0.8	5.26	15.92	79.8	0.65	0.65	65.9
Mix	6.58	26.98	51.9	0.58	0.55	45.7

(b) Auxiliary Loss Targets

Target	rFID↓	gFID↓	IS↑	Prec.↑	Rec.↑
DINO	4.00	12.89	91.0	0.70	0.66
D+HOG	*Training Collapsed*				
D+CLIP	**3.60**	**12.46**	**92.3**	**0.70**	**0.66**
C+HOG	4.18	20.77	56.0	0.63	0.59
All	3.64	12.82	88.6	0.70	0.66

(c) Token Number (K)

K	rFID↓	gFID↓	IS↑	Prec.↑	Rec.↑
64	5.61	12.78	91.8	**0.71**	0.65
128	**3.60**	**12.46**	**92.3**	0.70	**0.66**
192	24.59	30.42	68.0	0.57	0.57

(d) Latent Noise Scale (σ)

σ	rFID↓	gFID↓	IS↑	Prec.↑	Rec.↑
0.2	3.93	14.30	81.5	0.66	0.66
0.3	**3.40**	**12.82**	**88.6**	**0.70**	**0.66**
0.5	6.95	17.69	75.6	0.65	0.62

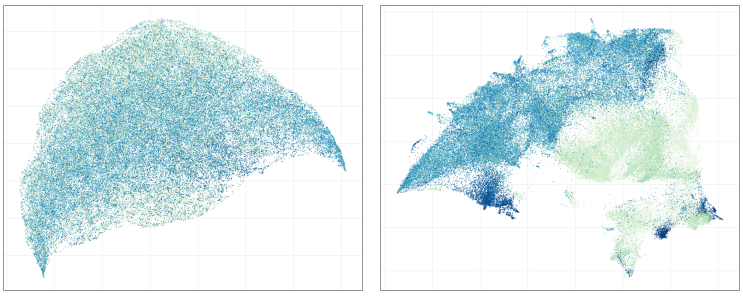
Comparison with Diffusion and AR Models. While Latent Diffusion Models (like DiT and REPA) currently lead in FID, MIMFlow-L narrows the gap between ODE-based Normalizing Flows and these heavy-duty generative frameworks. Unlike diffusion models that require hundreds of denoising steps or complex scheduling, our flow-based approach offers exact density estimation and a deterministic mapping in a single continuous ODE trajectory. The competitive Precision (0.82) of MIMFlow-L underscores its ability to generate high-fidelity samples that faithfully respect the learned data manifold.

In summary, the results demonstrate that by decoupling semantic modeling from texture synthesis, MIMFlow-L mitigates a capacity bottleneck in traditional NFs and provides an efficient path for improving NF image generation.

4.3 Ablation Study

To validate the effectiveness of the proposed components in MIMFlow, we conduct a series of ablation experiments on the ImageNet 256 × 256 benchmark. To ensure a fair comparison, all models in this section are trained end-to-end for 50 epochs. Unless otherwise specified, the reported generative metrics are calculated using 10K samples (50K in Tab. 3).

The Necessity of Masked Decoupling. Table 5a demonstrates the critical role of the masking strategy under the same end-to-end training setup. Without masking (*nomask*), the model exhibits the poorest performance (gFID 29.0), and its latent semantic quality (Acc. 56.6%) is significantly lower than that of masked versions. This supports our hypothesis that unmasked training leaves the Normalizing Flow exposed to redundant pixel-level details. A mask ratio of



(a) SD-VAE **(b)** MIMFlow

Fig. 4: UMAP visualization on ImageNet of the learned latent space from (a) SD-VAE; (b) MIMFlow. Colors indicate different classes. MIMFlow presents a more discriminative latent space.

0.4–0.6 achieves the best balance, significantly enhancing both generative quality (gFID 12.82) and semantic representation (Acc. 71.3%). Interestingly, a mixed strategy or low mask ratios lead to performance degradation, suggesting that a consistent and substantial information bottleneck is important for stabilizing the semantic manifold.

Optimal Bottleneck Size. As shown in Tab. 5c, the number of latent tokens K serves as a physical bottleneck for information compression. While 64 tokens provide reasonable performance, increasing to 128 tokens yields the best reconstruction (rFID 3.60) and generation results. This observation aligns with the information density provided by our masking strategy: since a 256×256 image is partitioned into $16 \times 16 = 256$ patches and approximately 50% of the patches are masked during training, the remaining information can be optimally represented by a compressed latent space of 128 tokens. However, expanding to 192 tokens leads to a sharp performance drop (gFID 30.42). This indicates that an excessively large latent space allows high-frequency noise to leak into the flow model, thereby violating the principled decoupling and hindering the NF's ability to model global structure.

Synergy of Auxiliary Semantic Priors. We investigate various auxiliary supervision signals (DINO, CLIP, HOG) in Tab. 5b. The combination of DINO and CLIP features achieves the superior performance, as they provide complementary structural and semantic guidance. In contrast, incorporating low-level features like HOG either leads to training collapse or suboptimal results. This validates that the MIMFlow latent space is inherently optimized for high-level semantic abstractions rather than local gradient textures.

Impact of Latent Stochasticity. The additive Gaussian noise σ is a crucial trick for NF training. As shown in Tab. 5d, $\sigma = 0.3$ is the optimal value. Smaller values ($\sigma = 0.2$) fail to sufficiently smoothen the manifold, while larger values ($\sigma = 0.5$) introduce excessive variance that compromises reconstruction fidelity (rFID 6.95), confirming that a precise calibration of latent stochasticity is vital for end-to-end flow modeling.

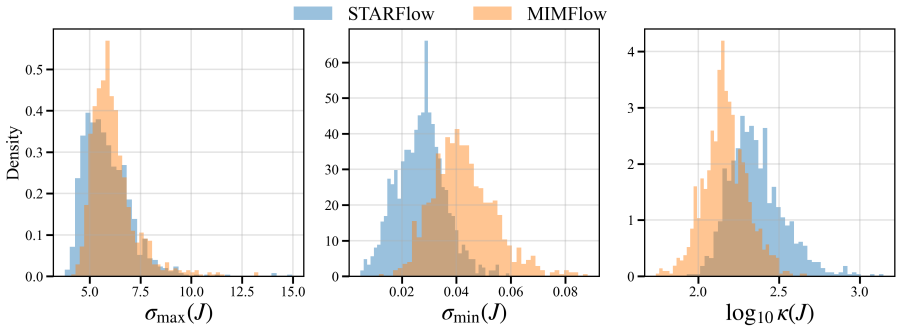


Fig. 5: Jacobian Spectral Analysis of STARFlow and MIMFlow. The three panels report, from left to right, the empirical distributions of the largest singular value $\sigma_{\max}(J)$, the smallest singular value $\sigma_{\min}(J)$, and the log-condition number $\log_{10}\kappa(J)$ (with $\kappa(J) = \sigma_{\max}(J)/\sigma_{\min}(J)$).

4.4 Analysis of Latent Space and Flow Dynamics

In this section, we further investigate why the proposed MIMFlow framework facilitates more effective generative modeling by analyzing the properties of the learned latent space and the flow transformations.

Semantic Discriminability. A core motivation of MIMFlow is to decouple high-frequency pixel noise from global structural semantics. We visualize the latent space $\mathbf{z}$ using UMAP projection in Fig. 4. Compared to the relatively entangled latent space of a standard SD-VAE, the MIMFlow latent space exhibits significantly clearer clustering corresponding to ImageNet categories. This enhanced discriminative power is quantitatively supported by the linear probing results in Tab. 5a: our masked approach achieves a classification accuracy of **71.3%**, a substantial improvement over the **56.6%** of the unmasked baseline. These results confirm that the masking bottleneck effectively forces the latent manifold to prioritize high-level semantic coherence over local redundancies.

Jacobian Spectral Analysis. To assess the numerical stability and bijectivity of the learned Normalizing Flow, we analyze the spectral properties of the Jacobian $J = \partial f_\theta / \partial \mathbf{z}$ for both STARFlow and MIMFlow (Fig. 5). The analysis reveals several key insights:

1. *Enhanced Bijectivity:* MIMFlow exhibits a larger and more stable minimum singular value ($\sigma_{\min}$), keeping the Jacobian further from singularity and reducing the risk of ill-conditioned mappings.
2. *Superior Numerical Stability:* Compared to STARFlow, MIMFlow achieves a lower and more concentrated log-condition number. A lower condition number indicates a more well-conditioned transformation that is less sensitive to numerical perturbations.

These spectral properties suggest that by alleviating the burden on the NF to capture pixel-level details, MIMFlow learns a smoother flow field with less

extreme spatial warping, which is consistent with more stable optimization and higher generative fidelity.

5 Limitation

Our experiments validate MIM-style masked semantic bottlenecks for latent NF models, but do not imply a general claim about MIM across all generative model families, such as diffusion or autoregressive models. This work also focuses on class-to-image generation on ImageNet; extending it to text-to-image generation, where prompt alignment and compositional semantics are central, remains future work. Finally, while we use standard ImageNet generation metrics such as FID, IS, precision/recall, and linear probing, newer metrics such as Representation Fréchet Loss [47] may provide complementary signals and are worth exploring.

6 Conclusion

In this paper, we presented MIMFlow, a unified end-to-end generative framework that integrates Masked Image Modeling with Normalizing Flows. By introducing a masked bottleneck with learnable tokens, we achieve a principled decoupling of generative tasks: the Normalizing Flow focuses on modeling the low-frequency semantic manifold, while a specialized decoder handles high-frequency texture synthesis. This design mitigates the capacity bottleneck of traditional NFs, which often spend representational power on redundant pixel-level noise. Empirical results on ImageNet 256×256 demonstrate that MIMFlow-L improves similar-scale NF baselines, achieving an FID of 2.50. Notably, our model achieves this with only 128 tokens–a 50% reduction compared to standard latent generative models–while maintaining high semantic discriminability, as evidenced by a 71.3% linear probing accuracy. Our analysis of the Jacobian spectrum further suggests that this decoupling leads to better-conditioned flow transformations. Ultimately, MIMFlow provides a new perspective on unifying self-supervised representation learning and probabilistic modeling, paving the way for more efficient and semantically-aware generative systems.

Acknowledgements

This work is supported by the Basic Research Program of Jiangsu (No. BK20250009), the Fundamental Research Funds for the Central Universities (No.020214380140), the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (No. JYB2025XDXM118), the Collaborative Innovation Center of Novel Software Technology and Industrialization, Alibaba Group through Alibaba Innovative Research Program.

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 (2025)
2. Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.: Neural ordinary differential equations (2019)
3. Chen, S., Ge, C., Zhang, S., Sun, P., Luo, P.: Pixelflow: Pixel-space generative models with flow. arXiv preprint arXiv:2504.07963 (2025)
4. Chen, Y., Xu, X., Wang, S., Zhu, C., Wen, R., Li, X., Ge, T., Wang, L.: Flowing backwards: Improving normalizing flows via reverse representation alignment. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 3074–3082 (2026)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-scale Hierarchical Image Database. IEEE Conference on Computer Vision and Pattern Recognition pp. 248–255 (2009)
6. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems **34**, 8780–8794 (2021)
7. Dinh, L., Krueger, D., Bengio, Y.: Nice: Non-linear independent components estimation. arXiv preprint arXiv:1410.8516 (2014)
8. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. arXiv preprint arXiv:1605.08803 (2016)
9. Draxler, F., Sorrenson, P., Zimmermann, L., Rousselot, A., Köthe, U.: Free-form flows: Make any architecture a normalizing flow. In: International Conference on Artificial Intelligence and Statistics. pp. 2197–2205. PMLR (2024)
10. Draxler, F., Wahl, S., Schnörr, C., Köthe, U.: On the universality of volume-preserving and coupling-based normalizing flows. arXiv preprint arXiv:2402.06578 (2024)
11. 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)
12. Gao, Y., Chen, C., Chen, T., Gu, J.: One layer is enough: Adapting pretrained visual encoders for image generation (2025)
13. Giaquinto, R., Banerjee, A.: Gradient boosted normalizing flows. Advances in Neural Information Processing Systems **33**, 22104–22117 (2020)
14. Gu, J., Chen, T., Berthelot, D., Zheng, H., Wang, Y., Zhang, R., Dinh, L., Bautista, M.A., Susskind, J., Zhai, S.: Starflow: Scaling latent normalizing flows for high-resolution image synthesis. arXiv preprint arXiv:2506.06276 (2025)
15. Gu, J., Chen, T., Shen, Y., Berthelot, D., Zhai, S., Susskind, J.: Normalizing trajectory models (2026)
16. Gu, J., Shen, Y., Chen, T., Dinh, L., Wang, Y., Bautista, M.A., Berthelot, D., Susskind, J., Zhai, S.: Starflow-v: End-to-end video generative modeling with autoregressive normalizing flows. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9084–9094 (2026)
17. Han, C., Fan, J., Wu, N., Dai, J., Bao, H., Lu, X.: Object-centric Video Prediction with Mask-guided Spatiotemporal Diffusion. Machine Intelligence Research pp. 1–11 (2026)
18. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners (2021)
19. 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)

20. Hoogeboom, E., Mensink, T., Heek, J., Lamerigts, K., Gao, R., Salimans, T.: Simpler diffusion (sid2): 1.5 fid on imagenet512 with pixel-space diffusion. arXiv preprint arXiv:2410.19324 (2024)
21. Hu, J., Wu, L., Chen, Y., Hu, P., Zaki, M.J.: GraphFlow+: Exploiting Conversation Flow in Conversational Machine Comprehension with Graph Neural Networks. Machine Intelligence Research **21**(2), 272–282 (2024)
22. Jabri, A., Fleet, D., Chen, T.: Scalable adaptive computation for iterative generation. arXiv preprint arXiv:2212.11972 (2022)
23. Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions (2018)
24. Kobyzev, I., Prince, S.J., Brubaker, M.A.: Normalizing flows: An introduction and review of current methods. IEEE transactions on pattern analysis and machine intelligence **43**(11), 3964–3979 (2020)
25. 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)
26. Lee, S.H., Park, S., Kim, G.M.: REPA-E: End-to-end training of latent-diffusion models via representation alignment. In: arXiv preprint arXiv:2405.18373 (2024)
27. Li, C., Tang, H., Zhu, Y., Yamanishi, Y.: A reinforcement learning-driven transformer gan for molecular generation. Machine Intelligence Research pp. 1–22 (2026)
28. Li, T., Tian, Y., Li, H., Deng, M., He, K.: Autoregressive image generation without vector quantization. Advances in Neural Information Processing Systems **37**, 56424–56445 (2024)
29. 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)
30. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
31. Papamakarios, G., Nalisnick, E., Rezende, D.J., Mohamed, S., Lakshminarayanan, B.: Normalizing flows for probabilistic modeling and inference. Journal of Machine Learning Research **22**(57), 1–64 (2021)
32. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4195–4205 (2023)
33. Qiu, Z., Wang, Z., Zheng, B., Huang, Z., Wen, K., Yang, S., Men, R., Yu, L., Huang, F., Huang, S., Liu, D., Zhou, J., Lin, J.: Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free (2025)
34. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020 (2021)
35. Ren, S., Yu, Q., He, J., Shen, X., Yuille, A., Chen, L.C.: Beyond next-token: Next-x prediction for autoregressive visual generation. arXiv preprint arXiv:2502.20388 (2025)
36. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: Bach, F., Blei, D. (eds.) Proceedings of the 32nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 37, pp. 1530–1538. PMLR, Lille, France (07–09 Jul 2015)
37. 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)

38. 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)
39. Shen, Y., Chen, T., Gao, Y., Zhang, Y., Wang, Y., Ángel Bautista, M., Zhai, S., Susskind, J.M., Gu, J.: Starflow2: Bridging language models and normalizing flows for unified multimodal generation (2026)
40. Singh, J., Zheng, B., Wu, Z., Zhang, R., Shechtman, E., Xie, S.: Improved baselines with representation autoencoders (2026)
41. Tian, K., Jiang, Y., Yuan, Z., Peng, B., Wang, L.: Visual autoregressive modeling: Scalable image generation via next-scale prediction. Advances in neural information processing systems **37**, 84839–84865 (2024)
42. Tschannen, M., Pinto, A.S., Kolesnikov, A.: Jetformer: An autoregressive generative model of raw images and text. arXiv preprint arXiv:2411.19722 (2024)
43. Tu, G., Fu, X., Yu, S., Tang, Y., Kang, H., Qin, L., Zhang, Y., Gu, J.: Latent reasoning with normalizing flows (2026)
44. Wang, S., Gao, Z., Zhu, C., Huang, W., Wang, L.: Pixnerd: Pixel neural field diffusion (2025)
45. Wang, S., Tian, Z., Huang, W., Wang, L.: Ddt: Decoupled diffusion transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 40633–40642 (June 2026)
46. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: Simmim: A simple framework for masked image modeling (2022)
47. Yang, J., Geng, Z., Ju, X., Tian, Y., Wang, Y.: Representation fréchet loss for visual generation. arXiv preprint arXiv:2604.28190 (2026)
48. Yang, J., Li, T., Fan, L., Tian, Y., Wang, Y.: Latent denoising makes good tokenizers (2026)
49. Yao, J., Song, Y., Zhou, Y., Wang, X.: Towards scalable pre-training of visual tokenizers for generation (2025)
50. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models (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. arXiv preprint arXiv:2410.06940 (2024)
52. Zhai, S., Zhang, R., Nakkiran, P., Berthelot, D., Gu, J., Zheng, H., Chen, T., Bautista, M.A., Jaitly, N., Susskind, J.: Normalizing flows are capable generative models. arXiv preprint arXiv:2412.06329 (2024)
53. Zhao, Q., Zheng, G., Yang, T., Zhu, R., Leng, X., Gould, S., Zheng, L.: Simflow: Simplified and end-to-end training of latent normalizing flows (2025)
54. Zheng, B., Ma, N., Tong, S., Xie, S.: Diffusion transformers with representation autoencoders (2025)
55. Zheng, G., Zhao, Q., Yang, T., Xiao, F., Lin, Z., Wu, J., Deng, J., Zhang, Y., Zhu, R.: Farmer: Flow autoregressive transformer over pixels (2025)
56. Zheng, H., Nie, W., Vahdat, A., Anandkumar, A.: Fast training of diffusion models with masked transformers. In: Transactions on Machine Learning Research (TMLR) (2024)
57. Zheng, Y., Tian, Y., Li, S., Wu, Z., Liu, B., Li, J., Ye, B., Zhou, J.R.: LightningDiT: A vision-foundation-model-aligned VAE for fast and high-quality generation. In: arXiv preprint arXiv:2405.15438 (2024)