Masked Autoencoders Are Effective Tokenizers for Diffusion Models

Hao Chen[*1,2] **Yujin Han**[*3] **Fangyi Chen**[1] **Xiang Li**[1] **Yidong Wang**[4]
Jindong Wang[5] **Ze Wang**[2] **Zicheng Liu**[2] **Difan Zou**[3] **Bhiksha Raj**[1]

Abstract

Recent advances in latent diffusion models have demonstrated their effectiveness for high-resolution image synthesis. However, the properties of the latent space from tokenizer for better learning and generation of diffusion models remain under-explored. Theoretically and empirically, we find that improved generation quality is closely tied to the latent distributions with better structure, such as the ones with fewer Gaussian Mixture modes and more discriminative features. Motivated by these insights, we propose **MAETok**, an autoencoder (AE) leveraging mask modeling to learn semantically rich latent space while maintaining reconstruction fidelity. Extensive experiments validate our analysis, demonstrating that the variational form of autoencoders is not necessary, and a discriminative latent space from AE alone enables state-of-the-art performance on ImageNet generation using only **128** tokens. MAETok achieves significant practical improvements, enabling a gFID of **1.69** with **76×** faster training and **31×** higher inference throughput for 512×512 generation. Our findings show that the structure of the latent space, rather than variational constraints, is crucial for effective diffusion models. Code and trained models are released[1].

1. Introduction

Diffusion models (Sohl-Dickstein et al., 2015a; Ho et al., 2020; Rombach et al., 2022a; Peebles & Xie, 2023) have recently emerged as a powerful class of generative models, achieving state-of-the-art (SOTA) performance on various image synthesis tasks (Deng et al., 2009; Ghosh et al., 2024). Although originally formulated in pixel space (Ho et al., 2020; Dhariwal & Nichol, 2021), subsequent research has shown that operating in a *latent space* – a compressed representation typically learned by a tokenizer – can substantially improve the efficiency and scalability of diffusion models (Rombach et al., 2022a). By avoiding the high-dimensional pixel domain during iterative diffusion and denoising steps, latent diffusion models dramatically reduce computational overhead and have quickly become the *de facto* paradigm for high-resolution generation (Esser et al., 2024).

However, a key question remains: *What constitutes a "good" latent space for diffusion*? Early work primarily employed *Variational Autoencoders* (VAE) (Kingma, 2013) as tokenizers, which ensure that the learned latent codes follow a relatively smooth distribution (Higgins et al., 2017) via a Kullback–Leibler (KL) constraint. While VAEs can empower strong generative results (Ma et al., 2024; Li et al., 2024b; Deng et al., 2024), they often struggle to achieve high pixel-level fidelity in reconstructions due to the imposed regularization (Tschannen et al., 2025). In contrast, recent explorations with *plain Autoencoders* (AE) (Hinton & Salakhutdinov, 2006; Vincent et al., 2008) produce higher-fidelity reconstructions but may yield latent spaces that are insufficiently organized or too entangled for downstream generative tasks (Chen et al., 2024b). Indeed, more recent studies emphasize that high fidelity to pixels does not necessarily translate into robust or semantically disentangled latent representations (Esser et al., 2021; Yao & Wang, 2025); leveraging latent alignment with pre-trained models can often improve generation performance further (Li et al., 2024c; Chen et al., 2024a; Qu et al., 2024; Zha et al., 2024).

In this work, we attempt to answer this question by investigating the interaction between *the latent distribution learned by tokenizers*, and *the training and sampling behavior of diffusion models* operating in that latent space. Specifically, we study AE, VAE and the recently emerging representation aligned VAE (Li et al., 2024c; Chen et al., 2024a; Zha et al., 2024; Yao & Wang, 2025), by fitting a Gaussian mixture model (GMM) into their latent space. Empirically, we show that a latent space with more *discriminative* features, whose GMM modes are *fewer*, tends to produce a lower diffusion loss. Theoretically, we prove that a latent distribution with fewer GMM modes indeed leads to a lower loss of diffusion

[*]Equal contribution [1]Carnegie Mellon University [2]AMD [3]The University of Hong Kong [4]Peking University [5]William & Mary. Correspondence to: Hao Chen <haoc3@andrew.cmu.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

[1]https://github.com/Hhhhhhao/continuous_tokenizer.

Figure 1. Diffusion models with MAETok achieves state-of-the-art image generation on ImageNet of 512×512 and 256×256 resolution.

models and thus to better sampling during inference.

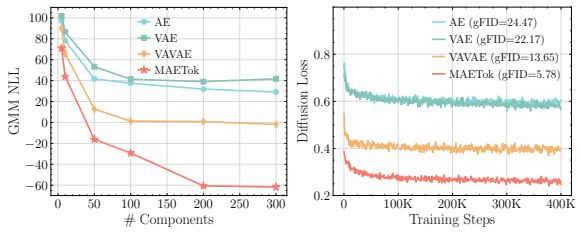
Motivated by these insights, we demonstrate that diffusion models trained on *AE*s with discriminative latent space are enough to achieve SOTA performance. We propose to train AEs as *Masked Autoencoders* (MAE) (He et al., 2022; Xie et al., 2022; Wei et al., 2022), a self-supervised paradigm that can discover more generalized and discriminative representations by reconstructing proxy features (Zhang et al., 2022). More specifically, we adopt the transformer architecture of tokenizers (Yu et al., 2021; 2024c; Li et al., 2024c; Chen et al., 2024a) and randomly mask the image tokens at the encoder, whose features need to be reconstructed at the decoder (Assran et al., 2023). To maintain a pixel decoder with high reconstruction fidelity, we adopt auxiliary shallow decoders that predict the features of unseen tokens from seen ones to learn the representations, along with the pixel decoder which is normally trained as previous tokenizers. The auxiliary shallow decoders introduce trivial computation overhead during training. This design allows us to extend the MAE objective that reconstructs masked image patches, to simultaneously predict *multiple targets*, such as HOG (Dalal & Triggs, 2005) features (Wei et al., 2022), DINOv2 features (Oquab et al., 2023), CLIP embeddings (Radford et al., 2021; Zhai et al., 2023), and Byte-Pair Encoding (BPE) indices with text (Huang et al., 2024).

Furthermore, we reveal an interesting decoupling effect: the capacity to learn a *discriminative and semantically rich* latent space at the encoder can be separated from the capacity to *achieve high reconstruction fidelity* at the decoder. In particular, a higher mask ratio (40–60%) in MAE training often degrades immediate pixel-level quality. However, by *freezing* the AE's encoder, thus preserving its well-organized latent space, and *fine-tuning only the decoder*, we can recover strong pixel-level reconstruction fidelity without sacrificing the semantic benefits of the learned representations.

Extensive experiments on ImageNet (Deng et al., 2009) demonstrate the effectiveness of MAETok. It addresses the trade-off between reconstruction fidelity and discriminative latent space by training the plain AEs with mask modeling, showing that the structure of latent space is more crucial for diffusion learning, instead of the variational forms of VAEs. MAETok achieves improved reconstruction FID (rFID) and generation FID (gFID) using only **128** tokens for the 256×256 and 512×512 ImageNet benchmarks.

Our contributions can be summarized as follows:

- **Theoretical and Empirical Analysis**: We establish a connection between latent space structure and diffusion model performance through both empirical and theoretical analysis. We reveal that structured latent spaces with fewer *Gaussian Mixture Model* modes enable more effective training and generation of diffusion models.
- **MAETok**: We train plain AEs using mask modeling and show that simple AEs with more discriminative latent space empower faster learning, better generation, and higher throughput of diffusion models, showing that the variational regularization of VAE is not necessary.
- **SOTA Generation Performance**: Diffusion models of 675M parameters trained on MAETok with 128 tokens achieve performance comparable to previous best models on 256 ImageNet generation and outperform previous 2B USiT at 512 resolution with 1.69 gFID and 304.2 IS.



(a) GMM Loss (b) Diffusion Loss

Figure 2. GMM fitting on latent space of AE, VAE, VAVAE, and MAETok. Fewer GMM modes in latent space usually corresponds to lower diffusion losses and better generation performance.

2. On the Latent Space and Diffusion Models

To study the relationship of latent space for diffusion models, we start with popular tokenizers, including AE (Hinton & Salakhutdinov, 2006), VAE (Kingma, 2013), representation aligned VAE, i.e., VAVAE (Yao & Wang, 2025). We train our own AE and VAE tokenizers under the same training recipe and the same dimension for fair comparison. We train diffusion models on them and establish connections between latent space properties and the quality of the final image generation through empirical and theoretical analysis.

Empirical Analysis. Inspired by existing theoretical work (Chen et al., 2022; 2023; Benton et al., 2024), our investigation of the connection between latent space and generation quality starts with a high-level intuition. With optimal diffusion model parameters, such as sufficient total time steps and adequately small discretization steps, and with assumed similar capacity of tokenizer decoders, the generation quality of diffusion models, i.e., the learned latent distribution, is dominated by the denoising network's training loss (Chen et al., 2022; 2023; Benton et al., 2024), while the effectiveness of training diffusion model via DDPM (Ho et al., 2020) heavily depends on the hardness of learning the latent space distribution (Shah et al., 2023; Diakonikolas et al., 2023; Gatmiry et al., 2024). Specially, when the training data distribution is too complex and multi-modal, i.e., not discriminative enough, the denoising network may struggle to capture such entangled global structure of latent space, resulting in a degraded generation quality.

Building upon this intuition, we use the *Gaussian Mixture Models* (GMM) to evaluate the number of modes in alternative latent space representations, where a higher number of modes indicates a more complex structure. The details of GMM training are included in Appendix B.3. Fig. 2a analyzes the GMM fitting by varying the number of Gaussian components and comparing their negative log-likelihood losses (NLL) across different latent spaces, where a lower NLL indicates better fitting quality. We observe that, to achieve comparable fitting quality, i.e., similar GMM losses, VAVAE requires fewer modes compared to VAE and AE. Fewer modes are sufficient to adequately represent the latent space distributions of VAVAE compared to those of AE and VAE, highlighting simpler global structures in its latent space. Correspondingly, Fig. 2b reports the training losses of diffusion models with AE, VAE, and VAVAE, which (almost) align with the GMM losses shown in Fig. 2a, where fewer modes correspond to lower diffusion losses and better gFID. This alignment validates our intuition, confirming that latent spaces with fewer modes and thus more separated and discriminative features can reduce the learning difficulty and lead to better generation quality of diffusion models.

Theoretical Analysis. After observing experimental phenomena that align with our high-level intuition, we further present a concise theoretical analysis here to justify the rationale behind it, with more details provided in Appendix A.

Following the empirical analysis setup, we first consider a latent data distribution in d dimensions modeled as a GMM with K equally weighted Gaussians:

$$p_0 = \frac{1}{K}\sum_{i=1}^{K}\mathcal{N}(\boldsymbol{\mu}_i^*, \mathbf{I}), \tag{1}$$

Considering the classic diffusion model DDPM (Ho et al., 2020) and following the training objective as Shah et al. (2023), the score matching loss of DDPM at timestep t is

$$\min_{\mathbf{w}} \mathbb{E}[\|s_{\mathbf{w}}(\mathbf{x}, t) - \nabla_{\mathbf{x}} \log p_t(\mathbf{x})\|^2], \tag{2}$$

where $s_{\mathbf{w}}(\mathbf{x}, t)$ represents the denoising network and $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ denotes the oracle score function.

Then, we establish the following theorem to show that more modes typically require larger training sample sizes for diffusion models to achieve comparable generation quality.

Theorem 2.1. *(Informal, see Theorem A.7) Let the data distribution be a mixture of K Gaussians as defined in Eq. (1). Then assume the norm of each mode is bounded by some constants, let d be the data dimension, T be the total time steps, and ϵ be a proper target error parameter. In order to achieve a $O(T\epsilon^2)$ error in KL divergence between data distribution and generation distirbution, the DDPM algorithm may require using $n \geq n'$ number of samples:*

$$n' = \Theta\left(\frac{K^4 d^5 B^6}{\varepsilon^2}\right), \tag{3}$$

where the upper bound of the mean norm satisfies $\max_i \|\boldsymbol{\mu}_i\| \leq B$.

Theorem 2.1 combines Theorem 16 from (Shah et al., 2023) and Theorem 2.2 from (Chen et al., 2023), showing that to achieve a comparable generation quality $O(T\epsilon^2)$, latent spaces with more modes (K) require a larger training sample size, scaling as $\mathcal{O}(K^4)$.This theoretically help explain

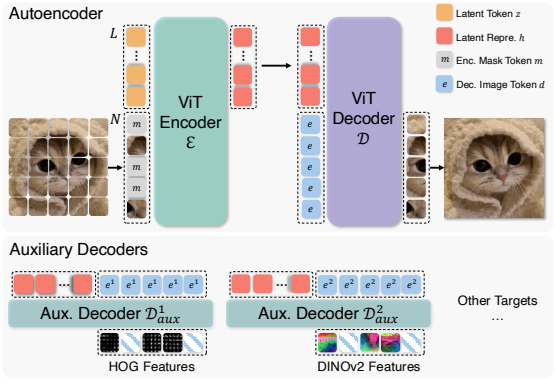


Figure 3. Model architecture of MAETok. We adopt the plain 1D autoencoder (AE) as tokenizer, with a vision transformer (ViT) encoder $\mathcal{E}$ and decoder $\mathcal{D}$. MAETok is trained using mask modeling at encoder, with a mask ratio of 40-60%, and predict multiple target features, e.g., HOG, DINO-v2, and CLIP features, of masked tokens from the unmasked ones using auxiliary shallow decoders.

why, under a finite number of training samples, latent spaces with more modes (e.g., AE and VAE) produce worse generations with higher gFID. We provide additional experimental results in Appendix A, demonstrating that these latent distributions share comparable upper bounds B, thus justifying our focus primarily on the impact of mode number K.

3. Method

Motivated by our analysis, we show that the variational form of VAEs may not be necessary for diffusion models, and simple AEs are enough to achieve SOTA generation performance with **128** tokens, as long as they have discriminative latent spaces, i.e., with fewer GMM modes. We term our method as **MAETok**, with more details as follows.

3.1. Architecture

We build MAETok upon the recent 1D tokenizer design with learnable latent tokens (Yu et al., 2024c; Li et al., 2024c; Chen et al., 2024a). Both the encoder $\mathcal{E}$ and decoder $\mathcal{D}$ adopt the Vision Transformer (ViT) architecture (Dosovitskiy et al., 2021; Yu et al., 2021), but are adapted to handle both image tokens and latent tokens, as shown in Fig. 3.

Encoder. The encoder first divides the input image $I \in \mathbb{R}^{H\times W\times 3}$ into N patches according to a predefined patch size P, each mapped to an embedding vector of dimension D, resulting in image tokens $\mathbf{x} \in \mathbb{R}^{N\times D}$. In addition, we define a set of L learnable latent tokens $\mathbf{z} \in \mathbb{R}^{L\times D}$. The encoder transformer takes the concatenation of image patch embeddings and latent tokens $[\mathbf{x};\mathbf{z}] \in \mathbb{R}^{(N+L)\times D}$ as its input, and outputs the latent representations $\mathbf{h} \in \mathbb{R}^{L\times H}$ with a dimension of H from only the latent tokens:

$$\mathbf{h} = \mathcal{E}\left([\mathbf{x};\mathbf{z}]\right). \tag{4}$$

Decoder. To reconstruct the image, we use a set of N learnable image tokens $\mathbf{e} \in \mathbb{R}^{N\times H}$. We concatenate these mask tokens with $\mathbf{h}$ as the input to the decoder, and takes only the outputs from mask tokens for reconstruction:

$$\hat{\mathbf{x}} = \mathcal{D}([\mathbf{e};\mathbf{h}]]). \tag{5}$$

We then use a linear layer on top of $\hat{\mathbf{x}} \in \mathbb{R}^{N\times D}$ to regress the pixel values and obtain the reconstructed image $\hat{I}$.

Position Encoding. To encode spatial information, we apply 2D Rotary Position Embedding (RoPE) to the image patch tokens $\mathbf{x}$ at the encoder and the image tokens $\mathbf{e}$ at the decoder. In contrast, the latent tokens $\mathbf{z}$ (and their encoded counterparts $\mathbf{h}$) use standard 1D absolute position embeddings, since they do not map to specific spatial locations. This design ensures that patch-based tokens retain the notion of 2D layout, while the learned latent tokens are treated as a set of abstract features within the transformer architecture.

Training objectives. We train MAETok using the standard tokenizer losses as in previous work (Esser et al., 2021):

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_1\mathcal{L}_{\text{percep}} + \lambda_2\mathcal{L}_{\text{adv}}, \tag{6}$$

with $\mathcal{L}_{\text{recon}}$, $\mathcal{L}_{\text{percep}}$, and $\mathcal{L}_{\text{adv}}$ denoting as pixel-wise mean-square-error (MSE) loss, perceptual loss (Larsen et al., 2016; Johnson et al., 2016; Dosovitskiy & Brox, 2016; Zhang et al., 2018), and adversarial loss (Goodfellow et al., 2020; Isola et al., 2018), respectively, and λ_1 and λ_2 being hyperparameters. Note that MAETok is a plain AE architecture, therefore, it does not require any variational loss between the posterior and prior as in VAEs, which simplifies training.

3.2. Mask Modeling

Token Masking at Encoder. A key property of MAETok is that we introduce mask modeling during training, following the principles of MAE (He et al., 2022; Xie et al., 2022), to learn a more discriminative latent space in a self-supervised way. Specifically, we randomly select a certain ratio, e.g., 40%-60%, of the image patch tokens according to a binary masking indicator $M \in \mathbb{R}^N$, and replace them with the learnable mask tokens $\mathbf{m} \in \mathbb{R}^D$ before feeding them into the encoder. *All* the latent tokens are maintained to more heavily aggregate information on the unmasked image tokens and used to reconstruct the masked tokens at the decoder output.

Auxiliary Shallow Decoders. In MAE, a shallow decoder (He et al., 2022) or a linear layer (Xie et al., 2022; Wei et al., 2022) is required to predict the target features, e.g., raw pixel values, HOG features, and features from pre-trained models, of the masked image tokens from the remaining ones. However, since our goal is to train MAE as tokenizers, the pixel decoder $\mathcal{D}$ needs to be able to reconstruct images in high fidelity. Thus, we keep $\mathcal{D}$ as a similar capacity to $\mathcal{E}$, and incorporate auxiliary shallow decoders to predict

additional feature targets, which share the same design as the main pixel decoder but with fewer layers. Formally, each auxiliary decoder $\mathcal{D}^j_{\text{aux}}$ takes the latent representations $\mathbf{h}$ and concatenate with their own $\mathbf{d}^j$ as inputs, and output $\hat{\mathbf{y}}^j$ as the reconstruction of their feature target $\mathbf{y}^j \in \mathbb{R}^{N\times D^j}$:

$$\hat{\mathbf{y}}^j = \mathcal{D}^j_{\text{aux}}([\mathbf{e}^j; \mathbf{h}]; \theta), \tag{7}$$

where D^j denotes the dimension of target features. We train these auxiliary decoders along with our AE using additional MSE losses at only the masked tokens according to the masking indicator M, similarly to Xie et al. (2022):

$$\mathcal{L}_{\text{mask}} = \sum_j \left\| M \otimes \left(\hat{\mathbf{y}}^j - \mathbf{y}^j\right)\right\|_2^2. \tag{8}$$

3.3. Pixel Decoder Fine-Tuning

While mask modeling encourages the encoder to learn a better latent space, high mask ratios can degrade immediate reconstruction. To address this, after training AEs with mask modeling, we *freeze* the encoder, thus preserving the latent representations, and *fine-tune* only the pixel decoder for a small number of additional epochs. This process allows the decoder to adapt more closely to frozen latent codes of clean images, recovering the details lost during masked training. We use the same loss as in Eq. (6) for pixel decoder fine-tuning and discard all auxiliary decoders in this stage.

4. Experiments

We conduct comprehensive experiments to validate the design choices of MAETok, analyze its latent space, and benchmark the generation performance to show its superiority.

4.1. Experiments Setup

Implementation Details of Tokenizer. We use XQ-GAN codebase (Li et al., 2024d) to train MAETok. We use ViT-Base (Dosovitskiy et al., 2021), initialized from scratch, for both the encoder and the pixel decoder, which in total have 176M parameters. We set $L = 128$ and $H = 32$ for latent space. Three MAETok variants are trained on 256×256 ImageNet (Deng et al., 2009), and 512×512 ImageNet, and a subset of 512×512 LAION-COCO (Schuhmann et al., 2022) for 500K iterations, respectively. In the first stage training with mask modeling on ImageNet, we adopt a mask ratio of 40-60% , set by ablation, and 3 auxiliary shallow decoders for multiple targets of HOG (Dalal & Triggs, 2005), DINO-v2-Large (Oquab et al., 2023), and SigCLIP-Large (Zhai et al., 2023) features. We adopt an additional auxiliary decoder for tokenizer trained on LAION-COCO, which predicts the discrete indices of text captions for the image using a BPE tokenizer (Cherti et al., 2023; Huang et al., 2024). Each auxiliary decoder has 3 layers also set by ablation. We set $\lambda_1 = 1.0$ and $\lambda_2 = 0.4$. For the pixel decoder fine-tuning, we linearly decrease the mask ratio from 60% to 0% over 50K iterations, with the same training loss. More training details of tokenizers are shown in Appendix B.1.

Implementation Details of Diffusion Models. We use SiT (Li et al., 2024a) and LightningDiT (Yao & Wang, 2025) for diffusion-based image generation tasks after training MAETok. We set the patch size of them to 1 and use a 1D position embedding, and follow their original training setting for other parameters. We use SiT-L of 458M parameters for the analysis and ablation study. For main results, we train SiT-XL of 675M parameters for 4M steps and LightningDiT for 400K steps on ImageNet of resolution 256 and 512. More details are provided in Appendix B.2.

Evaluation. For tokenizer evaluation, we report the reconstruction Frechet Inception Distance (rFID) (Heusel et al., 2017), peak-signal-to-noise ratio (PSNR), and structural similarity index measure (SSIM) on ImageNet and MS-COCO (Lin et al., 2014) validation set. For the latent space evaluation of the tokenizer, we conduct linear probing (LP) on the flatten latent representations and report accuracy. To evaluate the performance of generation tasks, we report generation FID (gFID), Inception Score (IS) (Salimans et al., 2016), Precision and Recall (Kynkäänniemi et al., 2019) (in Appendix C.1), with and without classifier-free guidance (CFG) (Ho & Salimans, 2022), using 250 inference steps.

4.2. Design Choices of MAETok

We first present an extensive ablation study to understand how mask modeling and different designs affect the reconstruction of tokenizer and, more importantly, the generation of diffusion models. We start with an AE and add different components to study both rFID of AE and gFID of SiT-L.

Mask Modeling. In Table 1a, we compare AE and VAE with mask modeling and also study the proposed fine-tuning of the pixel decoder. For AE, mask modeling significantly improves gFID and slightly deteriorates rFID, which can be recovered through the decoder fine-tuning stage without sacrificing generation performance. In contrast, mask modeling only marginally improves the gFID of VAE, since the imposed KL constraint may hinder latent space learning.

Reconstruction Target. In Table 1b, we study how different reconstruction targets affect latent space learning in mask modeling. We show that using the low-level reconstruction features, such as the raw pixel (with only a pixel decoder) and HOG features, can already learn a better latent space, resulting in a lower gFID. Adopting semantic teachers such as DINO-v2 and CLIP instead can significantly improve gFID. Combining different reconstruction targets can achieve a balance in reconstruction fidelity and generation quality.

Mask Ratio. In Table 1c, we show the importance of proper mask ratio for learning the latent space using HOG target,

case	rFID	gFID
VAE	1.22	22.17
+MM	1.75	18.17
AE	0.67	24.47
+MM	0.85	5.78
+FT	0.48	5.69

(a) Mask modeling.

case	rFID	gFID
pixel	1.15	17.18
HOG	2.43	13.54
DINO	0.89	6.24
CLIP	0.78	11.31
Comb.	0.85	5.78

(b) Reconstruction target.

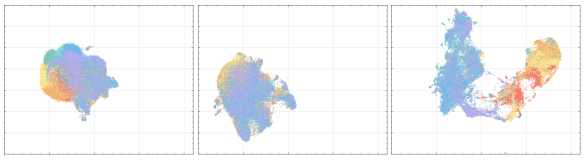
low	high	rFID	gFID
0	60	0.82	24.15
10	40	1.01	22.63
20	60	1.44	20.35
40	40	1.78	18.27
40	60	2.43	17.18

(c) Mask ratio (HOG w/o FT).

blocks	rFID	gFID
linear	1.35	6.98
1	1.19	6.43
3	0.85	5.78
6	0.86	7.12
12	0.96	8.80

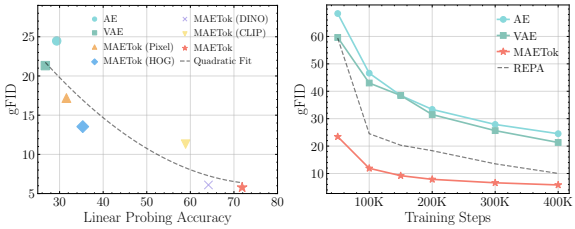
(d) Aux. decoder depth.

Table 1. Ablations with MAETok on 256×256 ImageNet. We report rFID of tokenizer and gFID of SiT-L trained on latent space of the tokenizer without classifier-free guidance. We train tokenizer of 250K and SiT-L for 400K steps. Default settings are indicated in Grey.



(a) AE (b) VAE (c) MAETok

Figure 4. UMAP visualization on ImageNet of the learned latent space from (a) AE; (b) VAE; (c) MAETok. Colors indicate different classes. MAETok presents a more discriminative latent space.



(a) gFID vs. LP Acc. (b) gFID during training

Figure 5. The latent space from tokenizer correlates strongly with generation performance. More discriminative latent space (a) with higher linear probing (LP) accuracy usually leads to better gFID, and (b) makes the learning of the diffusion model easier and faster.

as highlighted in previous works (He et al., 2022; Wei et al., 2022; Xie et al., 2022). A low mask ratio prevents the AE from learning more discriminative latent space. A high mask ratio imposes a trade-off between reconstruction fidelity and the latent space quality, and thus generation performance.

Auxiliary Decoder Depth. We study the depth of auxiliary decoder in Table 1d with multiple reconstruction targets. We show that a decoder that is too shallow or too deep could hurt both the reconstruction fidelity and generation quality. When the decoder is too shallow, the combined target features may confuse the latent with high-level semantics and low-level details, resulting in a worse reconstruction fidelity. However, a deeper auxiliary decoder may learn a less discriminative latent space of the AE with its strong capacity, and thus also lead to worse generation performance.

We include more ablation study on the number of learnable latent tokens and 2D RoPE in Appendix C.4.

4.3. Latent Space Analysis

We further analyze the relationship between the latent space of the AE variants and the generation performance of SiT-L.

Latent Space Visualization. We provide a UMAP visualization (McInnes et al., 2018) in Fig. 4 to intuitively compare the latent space learned by different variants of AE. Notably, both the AE and VAE exhibit more entangled latent embeddings, where samples corresponding to different classes tend to overlap substantially. In contrast, MAETok shows distinctly separated clusters with relatively clear boundaries between classes, suggesting that MAETok learns more discriminative latent representations. In line with our analysis in Section 2 and Fig. 2, a more discriminative and separated latent representation of MAETok results in much fewer GMM modes and improve the generation performance. More visualization is shown in Appendix C.3.

Latent Distribution and Generation Performance. We assess the latent space's quality by studying the relationship between the linear probing (LP) accuracy on the latent space, as a proxy of how well semantic information is preserved in the latent codes, and the gFID for generation performance. In Fig. 5a, we observe tokenizers with more discriminative latent distributions, as indicated by higher LP accuracy, correspondingly achieve lower gFID. This finding suggests that when features are well-clustered in latent space, the generator can more easily learn to generate high-fidelity samples. We further verify this intuition by tracking gFID throughout training, shown in Fig. 5b, where MAETok enables faster convergence, with gFID rapidly decreasing with lower values than the AE or VAE baselines. A high-quality latent distribution is shown to be a crucial factor in both achieving strong final generation metrics and accelerating training.

4.4. Main Results

Generation. We compare SiT-XL and LightningDiT based on variants of MAETok in Tables 2 and 3 for the 256×256 and 512×512 ImageNet benchmarks, respectively, against other SOTA generative models. Notably, the **naive SiT-XL** trained on MAETok with only **128 tokens and plain AE architecture** achieves consistently better gFID and IS without using CFG: it outperforms REPA (Yu et al., 2024d) by **3.59** gFID on 256 resolution and establishes a SOTA com-

Model (G)	# Params (G)	Model (T)	# Params (T)	# Tokens ↓	rFID ↓	w/o CFG gFID ↓	w/o CFG IS ↑	w/ CFG gFID ↓	w/ CFG IS ↑
Auto-regressive									
VQGAN (Esser et al., 2021)	1.4B	VQ	23M	256	7.94	–	–	5.20	290.3
ViT-VQGAN (Yu et al., 2021)	1.7B	VQ	64M	1024	1.28	4.17	175.1	–	–
RQ-Trans. (Lee et al., 2022)	3.8B	RQ	66M	256	3.20	–	–	3.80	323.7
MaskGIT (Chang et al., 2022)	227M	VQ	66M	256	2.28	6.18	182.1	–	–
LlamaGen-3B (Sun et al., 2024)	3.1B	VQ	72M	576	2.19	–	–	2.18	263.3
TiTok-S-128 (Yu et al., 2024c)	287M	VQ	72M	128	1.61	–	–	1.97	281.8
VAR (Tian et al., 2024)	2B	MSRQ†	109M	680	0.90	–	–	1.92	323.1
ImageFolder (Li et al., 2024c)	362M	MSRQ	176M	286	0.80	–	–	2.60	295.0
MAGVIT-v2 (Yu et al., 2024a)	307M	LFQ	116M	256	1.61	3.07	213.1	1.78	319.4
MaskBit (Weber et al., 2024)	305M	LFQ	54M	256	1.61	–	–	1.52	328.6
MAR-H (Li et al., 2024b)	943M	KL	66M	256	1.22	2.35	227.8	1.55	303.7
Diffusion-based									
LDM-4 (Rombach et al., 2022b)	400M	KL†	55M	4096	0.27	10.56	103.5	3.60	247.7
U-ViT-H/2 (Bao et al., 2023)	501M					–	–	2.29	263.9
MDTv2-XL/2 (Gao et al., 2023)	676M					5.06	155.6	1.58	314.7
DiT-XL/2 (Peebles & Xie, 2023)	675M	KL†	84M	1024	0.62	9.62	121.5	2.27	278.2
SiT-XL/2 (Ma et al., 2024)	675M					8.30	131.7	2.06	270.3
+ REPA (Yu et al., 2024d)						5.90	157.8	1.42	305.7
TexTok-256 (Zha et al., 2024)	675M	KL	176M	256	0.69	–	–	1.46	303.1
LightningDiT (Yao & Wang, 2025)	675M	KL	70M	256	0.28	2.17	205.6	1.35	295.3
Ours									
MAETok + LightningDiT	675M	AE	176M	**128**	0.48	2.21	208.3	1.73	308.4
MAETok + SiT-XL	675M					2.31	216.5	1.67	311.2

Table 2. System-level comparison on ImageNet 256×256 conditional generation. SiT-XL and LightningDiT trained on MAETok achieves performance comparable to state-of-the-art using plain AE with only 128 tokens. "Model (G)": the generation model. "# Params (G)": the number of generator's parameters. "Model (T)": the tokenizer model. "# Params (T)": the number of tokenizer's parameters. "# Tokens": the number of latent tokens used during generation. † indicates that the model has been trained on other data than ImageNet.

Model (G)	# Params (G)	Model (T)	# Params (T)	# Tokens ↓	rFID ↓	w/o CFG gFID ↓	w/o CFG IS ↑	w/ CFG gFID ↓	w/ CFG IS ↑
GAN									
BigGAN (Chang et al., 2022)	–	–	–	–	–	–	–	8.43	177.9
StyleGAN-XL (Karras et al., 2019)	168M	–	–	–	–	–	–	2.41	267.7
Auto-regressive									
MaskGIT (Chang et al., 2022)	227M	VQ	66M	1024	1.97	7.32	156.0	–	–
TiTok-B-64 (Yu et al., 2024c)	177M	VQ	202M	128	1.52	–	–	2.13	261.2
MAGVIT-v2 (Yu et al., 2024a)	307M	LFQ	116M	1024	-	–	–	1.91	324.3
MAR-H (Li et al., 2024b)	943M	KL	66M	1024	–	2.74	205.2	1.73	279.9
Diffusion-based									
ADM (Dhariwal & Nichol, 2021)	–	–	–	–	–	23.24	58.06	3.85	221.7
U-ViT-H/4 (Bao et al., 2023)	501M					–	–	4.05	263.8
DiT-XL/2 (Peebles & Xie, 2023)	675M	KL†	84M	4096	0.62	9.62	121.5	3.04	240.8
SiT-XL/2 (Ma et al., 2024)	675M					–	–	2.62	252.2
DiT-XL (Chen et al., 2024b)	675M					9.56	–	2.84	–
UViT-H (Chen et al., 2024b)	501M					9.83	–	2.53	–
UViT-H (Chen et al., 2024b)	501M	AE†	323M	256	0.22	12.26	–	2.66	–
UViT-2B (Chen et al., 2024b)	2B					6.50	–	2.25	–
USiT-2B (Chen et al., 2024b)	2B					2.90	–	1.72	–
Ours									
MAETok + LightningDiT	675M					2.56	224.5	1.72	307.3
MAETok + SiT-XL	675M	AE	176M	**128**	0.62	2.79	204.3	**1.69**	304.2
MAETok + USiT-2B	2B					**1.72**	**244.3**	**1.65**	**312.5**

Table 3. System-level comparison on ImageNet 512×512 conditional generation. SiT-XL and LightningDiT trained on MAETok achieve state-of-the-art performance using plain AE with only 128 tokens, outperforming USiT of 2B parameters using only 675M parameters.

parable gFID of **2.79** at 512 resolution. When using CFG, SiT-XL achieves a comparable performance with competing autoregressive and diffusion-based baselines trained on VAEs at 256 resolution. It beats the 2B USiT (Chen et al., 2024b) with 256 tokens and also achieves a new SOTA of **1.69** gFID and **304.2** IS at 512 resolution. Better results have been observed with LightningDiT, trained with more advanced tricks (Yao & Wang, 2025), where it outperforms MAR-H of 1B parameters and USiT of 2B parameters without CFG, achieves a **2.56** gFID and **224.5** IS, and **1.72** gFID with CFG. When using a USiT-2B (Chen et al., 2024b) for 512 generation, it pushes the gFID without CFG to **1.72**,