NESTOK: NESTED SELF-ALIGNED 1D TOKENIZER FOR AUTOREGRESSIVE IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

One-dimensional (1D) variable-length visual tokenizers enable adaptive compression by varying the number of tokens, allowing downstream autoregressive (AR) models to flexibly trade off generation quality against computational cost using a single tokenizer. However, existing approaches based on nested dropout often fail to fully exploit the representational capacity of the tokenizer, resulting in suboptimal performance in both image reconstruction and generation. In this work, we introduce NesTok, a nested self-alignment framework tailored to dynamic visual tokenizers. NesTok introduces cross-length training, which jointly optimizes reconstruction across token lengths while using the full-length sequence to guide shorter counterparts, enabling shorter token sequences to approach the reconstruction quality of full-length sequences. On ImageNet, NesTok improves substantially over standard training and achieves an rFID score of 0.98. On downstream image generation, it achieves the state-of-the-art gFID score of **1.46** on ImageNet 256×256 among existing variable-length autoregressive image generation methods.

1 INTRODUCTION

Image generation has achieved remarkable success under both diffusion-based and autoregressive (AR) paradigms (Ho et al., 2020; Li et al., 2026; Lee et al., 2022a; Sun et al., 2024b; Lu et al., 2026), with visual tokenizers playing a pivotal role in this progress. Recent studies (Yu et al., 2024b; He et al., 2025; Kim et al., 2025; Xiong et al., 2025) have explored one-dimensional (1D) visual tokenization to achieve higher compression ratios while preserving reconstruction fidelity. This formulation is conceptually closer to tokenization in natural language and has attracted increasing attention. However, most existing 1D tokenizers operate at a fixed token length and cannot adapt compression ratios within a single model. Supporting different computational budgets and target quality levels therefore requires multiple model variants.

Dynamic visual tokenizers address this limitation by mapping images to variable-length latent sequences of continuous embeddings or discrete codes (Miwa et al., 2025; Bachmann et al., 2025; Wen et al., 2025; Gao & Shou, 2026; Huang et al., 2025; Duggal et al., 2025; Wang et al., 2025), allowing the token budget to be adjusted according to computational constraints or image complexity. A prevalent strategy is nested dropout (Bachmann et al., 2025), which randomly truncates the latent sequence and remains the prefix during training, encouraging an ordered representation in which early tokens capture the most important visual information. Representative approaches include FlexTok (Bachmann et al., 2025), Semanticist (Wen et al., 2025) and One-D-Piece (Miwa et al., 2025). When shorter representations correspond to prefixes of the full-length sequence, a downstream AR generator can be trained exclusively on full-length sequences and generate prefixes of different lengths at inference time, enabling flexible trade-offs between generation quality and computational cost.

However, flexibility in token length is not necessarily equivalent to effective use of additional tokens. Prior studies (Fu et al., 2026; He et al., 2025; Bachmann et al., 2025; Wen et al., 2025) have shown that increasing sequence length may yield diminishing improvements in downstream AR generation quality or even degrade it. This observation suggests *inefficient information allocation* across token lengths: shorter prefixes may provide insufficient representations, while later tokens may fail to contribute useful complementary information. Effective variable-length tokenization therefore

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

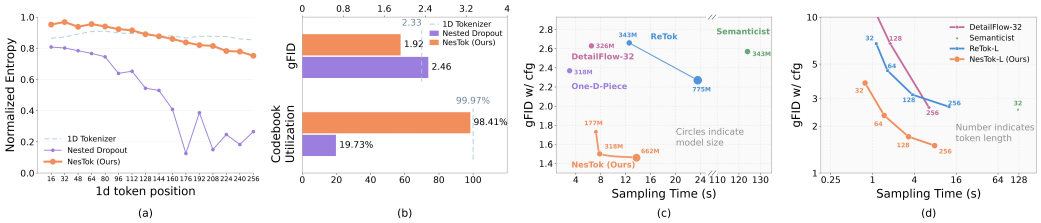


Figure 1: **Codebook statistics and AR generation performance on ImageNet-1K.** (a): Normalized entropy of the code distribution at each token position, computed over the training dataset and averaged within groups of 16 positions. (b): Average codebook utilization across token positions and gFID. (c): Sampling efficiency with different methods. (d): Sampling efficiency for different token lengths with different methods.

requires both informative prefixes and progressive refinement through additional tokens. To improve information allocation, ReTok (Fu et al., 2026) introduces redundant token padding, whereas CaTok (Chen et al., 2026) couples token selection intervals with time intervals in the MeanFlow objective to encourage a more balanced distribution of information.

We analyze codebook statistics and AR generation performance on ImageNet-1K, as shown in Fig. 1. Fig. 1 (a) shows that training with nested dropout results in substantially reduced code diversity at later token positions, reflected in their low normalized entropy. The result suggests underutilization of later tokens, potentially limiting their ability to complement the early tokens. Fig. 1 (b) further shows that nested dropout achieves a gFID of only 2.46, suggesting that concentrated code distribution and tail-token collapse lead to poor generation quality.

Beyond information allocation, decoder design also affects the simplicity and efficiency of variable-length image generation. Recent work has explored diffusion or flow-based decoders (Bachmann et al., 2025; Wen et al., 2025; Gao & Shou, 2026) to improve generation quality, potentially introducing additional architectural complexity and decoding latency. We focus on discrete variable-length tokenizers with ViT decoders, retaining the standard next-token prediction paradigm for downstream autoregressive generation. Within this setting, a central challenge is to learn representations that remain effective across token lengths: shorter prefixes should capture global visual content, while later tokens should contribute details for progressive refinement. Meeting these requirements calls for training strategies that coordinate learning across token lengths while supporting downstream AR modeling. This motivates us to revisit the representations learned by variable-length tokenizers: ***How can a single tokenizer learn informative prefixes and useful refinements that together support effective autoregressive generation?***

In this work, we introduce NesTok, a variable-length 1D tokenizer built on ViT and trained through *nested self-alignment learning*, which substantially improves autoregressive image generation. We first establish a training recipe for learning a variable-length tokenizer from scratch, without relying on any external distillation. Building on this recipe, we propose a *nested self-aligned tokenizer* that jointly optimizes reconstruction across token lengths while aligning the semantic representations of shorter sequences with those of their full-length counterparts. At each training iteration, representations associated with shorter token sequences are encouraged to align with the full-length sequence in latent space. We empirically demonstrate that this mechanism alleviates the *collapse of tail-token information* associated with nested dropout. Fig. 1 (a) shows that NesTok preserves code diversity across token positions, with a gradual entropy decline consistent with progressive refinement. In Fig. 1 (b), NesTok increases average codebook utilization from 19.73% to 98.41% and reduces gFID from 2.46 to 1.92, outperforming the fixed-length baseline (2.33). Fig. 1 (c) demonstrates a favorable trade-off between quality and latency across model sizes. With a single AR generator, NesTok also consistently improves generation quality as token length increases, achieving lower gFID at comparable sampling times (Fig. 1 (d)).

We validate the effectiveness of NesTok on ImageNet-1K 256×256 image generation. NesTok achieves a reconstruction FID (rFID) of **0.98** with 256 latent tokens (vs. 1.93 for our variable-length 1D baseline). When paired with LlamaGen, NesTok achieves a generation FID (gFID) of **1.46** with vanilla AR models, without introducing additional diffusion components. Furthermore, generation quality improves consistently as the number of tokens increases across token lengths. We

hope that this work encourages further exploration of ViT-based variable-length visual tokenizers for autoregressive image generation. Our contributions are as follows:

- We introduce **NesTok**, a variable-length 1D ViT tokenizer trained without external distillation to support flexible reconstruction and effective autoregressive generation.
- We propose **cross-length joint training** and **nested self-alignment** to improve token utilization and establish a coarse-to-fine ordering of visual information.
- NesTok achieves an rFID of **0.98** with 256 tokens and a gFID of **1.46** on ImageNet-1K 256×256 using vanilla AR generation. A trained generator supports different token lengths, with generation quality improving consistently as more tokens are generated.

2 Related Work

Image Tokenizers. Image tokenizers compress high-dimensional images into compact continuous or discrete latent representations for efficient generative modeling. Variational autoencoders (VAEs) (Kingma & Welling, 2013; Zeng et al., 2026; Yao et al., 2025) map images into continuous latent spaces and are optimized using a reconstruction loss with KL-divergence regularization. VQ-VAE (Van Den Oord et al., 2017; Razavi et al., 2019) and VQGAN (Esser et al., 2021) instead encode images into fixed 2D grids of discrete tokens. Subsequent studies (Yu et al., 2023; 2024a; Mentzer et al., 2024; Lee et al., 2022b; Shi et al., 2025) introduce advanced quantization techniques to improve reconstruction fidelity, codebook utilization, and downstream generation performance. Recently, 1D tokenizers (Yu et al., 2024b; Chen et al., 2025; Qu et al., 2026) have transformed images into compact token sequences, substantially reducing the spatial redundancy inherent in conventional 2D grids. The latest work, EOSTok (Chu et al., 2026) unifies the conventional two-stage tokenizer and generator pipeline through end-to-end training.

1D Variable-length Visual tokenizers. Compared with fixed-length 1D tokenizers, flexible visual tokenizers aim to encode images into variable-length token sequences using a single model (Bachmann et al., 2025; Miwa et al., 2025; Huang et al., 2025). Existing approaches can be categorized by their decoding mechanisms: **1) ViT decoders**: One-D-Piece (Miwa et al., 2025) employs nested tail dropping together with two-stage training and external distillation. SpectralAR (Huang et al., 2025) introduces a spectral information loss to encourage causal structure in 1D token sequences. ReTok (Fu et al., 2026) improves dynamic tokenization through redundant tokens and hierarchical semantic regularization. **2) Diffusion/Flow Decoders**: FlexTok (Bachmann et al., 2025) replaces the ViT decoder with a rectified-flow decoder. FlexTok achieves the best gFID with 32 tokens, while generating additional tokens degrades generation quality. D-AR (Gao & Shou, 2026) and CaTok (Chen et al., 2026) associate time steps in the decoding process with intervals of the token sequence to enable efficient image reconstruction. Despite these advances, unbalanced information allocation across token lengths remains a challenge, limiting the benefits of variable-length tokenization for downstream autoregressive generation. We propose a nested self-aligned training framework, which requires no external distillation and aims to alleviate this imbalance while improving ordered representations suitable for autoregressive modeling.

Modern Autoregressive Visual Generation. Inspired by GPT-style language models, early autoregressive image generators built on discrete visual tokenizers, such as VQ-VAE and VQGAN (Sun et al., 2024a; Van Den Oord et al., 2017; Esser et al., 2021), flatten two-dimensional token grids into one-dimensional sequences and generate tokens in raster-scan order using causal attention and next-token prediction (NTP). However, this fixed ordering restricts the use of bidirectional spatial context. MAR (Li et al., 2024) and MaskGIT (Yu et al., 2023) employ masked prediction with bidirectional attention to incorporate context from visible tokens during iterative generation. VAR (Tian et al., 2024) adopts next-scale prediction, reformulating image generation as a coarse-to-fine process that progressively introduces finer visual details. RAR (Yu et al., 2025) and RandAR (Pang et al., 2025) further explore randomized token prediction orders to improve contextual modeling. Although effective, these approaches introduce changes to the generation order, attention pattern, or prediction target of vanilla autoregressive modeling. Unlike these approaches that adapt the generator to visual data, we focus on reshaping the visual representations themselves. Our method encourages more balanced information allocation across token lengths while improving the compatibility of visual tokens with a vanilla autoregressive model without requiring additional changes to the generation mechanism.

3 METHOD

1D tokenizers have attracted considerable attention for their flexible compression capabilities. However, enabling efficient downstream autoregressive modeling with variable-length tokenizers remains challenging. Our goal is to learn variable-length token sequences in which short prefixes provide effective image representations and additional tokens contribute complementary refinements. To this end, NesTok combines cross-length joint training with nested self-alignment. These mechanisms aim to improve information utilization and encourage a coarse-to-fine ordering that enhances the *autoregressive compatibility* of the learned tokens. We first introduce the architecture of our ViT variable-length 1D tokenizer (Section 3.1), then present our nested self-alignment training method (Section 3.2), and finally introduce variable-length autoregressive modeling (Section 3.3).

3.1 1D VISION TOKENIZER AND AUTOREGRESSIVE MODEL

Our 1D variable-length tokenizer is based on One-D-Piece (Miwa et al., 2025), an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times C}$ is partitioned into non-overlapping patches of size $f \times f$ and mapped to a sequence of patch embeddings $\mathbf{x}_{\mathrm{p}} \in \mathbb{R}^{N\times D}$, where $N = HW/f^2$, C is the number of image channels, and $f = 16$. These embeddings are concatenated with latent tokens $\mathbf{q} \in \mathbb{R}^{K\times D}$ and processed by the encoder $\mathcal{E}_\psi$ to obtain $[\mathbf{h}_\mathcal{E}, \mathbf{z}] = \mathcal{E}_\psi([\mathbf{x}_{\mathrm{p}}, \mathbf{q}])$. We discard the encoded patch representations $\mathbf{h}_\mathcal{E}$ and retain the 1D latent representations $\mathbf{z}$, which are subsequently quantized as $\mathbf{z}_q = \mathcal{Q}(\mathbf{z})$ by looking up the closest entry using a vector quantizer. The quantized latent tokens are then concatenated with mask tokens $\mathbf{m}_{\mathrm{p}} \in \mathbb{R}^{N\times D}$ and passed to the decoder $\mathcal{D}_\phi$ for reconstruction, yielding $[\varnothing, \mathbf{x}_r] = \mathcal{D}_\phi([\mathbf{z}_q, \mathbf{m}_{\mathrm{p}}])$. Here, $\mathbf{m}_{\mathrm{p}}$ is formed by repeating a mask token N times, and $\varnothing$ denotes the discarded latent tokens.

Consider a variable-length 1D tokenizer that encodes an image into a quantized token sequence $\mathbf{z}_q = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_K]$ using an encoder $\mathcal{E}$ and a quantizer $\mathcal{Q}$. Here, K denotes the full sequence length, and $\mathbf{z}_k$ is the token at position k. During training, we uniformly sample an integer $k \in \{1, \ldots, K\}$ and truncate the sequence to retain a prefix of length k, denoted by $\mathbf{z}_q^{(k)} = [\mathbf{z}_1, \ldots, \mathbf{z}_k]$. The retained prefix is then passed to the decoder $\mathcal{D}$ to reconstruct the input image. For the discrete tokenizer considered here, we use a composite training objective comprising four terms (Yu et al., 2024c):

$$\mathcal{L}_{\mathrm{rec}} = \mathcal{L}_{\mathrm{mse}} + \mathcal{L}_{\mathrm{perc}} + \mathcal{L}_{\mathrm{quant}} + \mathcal{L}_{\mathrm{adv}}. \tag{1}$$

Here, $\mathcal{L}_{\mathrm{mse}}$ measures the mean squared error between the reconstructed image $\hat{\mathbf{x}}$ and the input image $\mathbf{x}$. The perceptual loss $\mathcal{L}_{\mathrm{perc}}$ combines LPIPS with a feature perceptual loss computed using ConvNeXt-S. The quantization loss $\mathcal{L}_{\mathrm{quant}}$ comprises the codebook and commitment terms, while $\mathcal{L}_{\mathrm{adv}}$ is an adversarial loss that encourages visually realistic reconstructions. In addition to gradient-based updates driven by the codebook loss, we apply auxiliary usage-adaptive codebook updates guided by exponential moving average (EMA) estimates of codebook usage (Zheng & Vedaldi, 2023).

Although most discrete visual tokenizers are trained with this form Wu et al. (2025), their performance is highly sensitive to the training recipe. We therefore establish a standard training recipe as our baseline, with details provided in Section 4.

Autoregressive Compatibility. The inherent spatial continuity of images introduces redundancy across neighboring patches. Although 1D ViT tokenizers reorganize spatial features into a latent sequence, their encoders jointly optimize token representations for reconstruction without explicitly encouraging an information ordering suitable for autoregressive prediction. Consequently, strong reconstruction performance does not necessarily imply strong *autoregressive compatibility*. Nested dropout encourages reconstruction from shorter prefixes, but later tokens are supplied to the decoder less frequently, potentially contributing to unbalanced information allocation and limiting the benefits of larger token budgets. Therefore, we propose a nested self-alignment training framework to jointly improve information allocation and token ordering (Section 3.2).

3.2 NESTED SELF-ALIGNMENT LEARNING

Cross-Length Joint Training. We introduce a cross-length joint training strategy to improve training across token lengths. For an input image $\mathbf{x}$, let $\mathbf{z}_q^{(K)} = [\mathbf{z}_1, \ldots, \mathbf{z}_K]$ denote full-length quantized

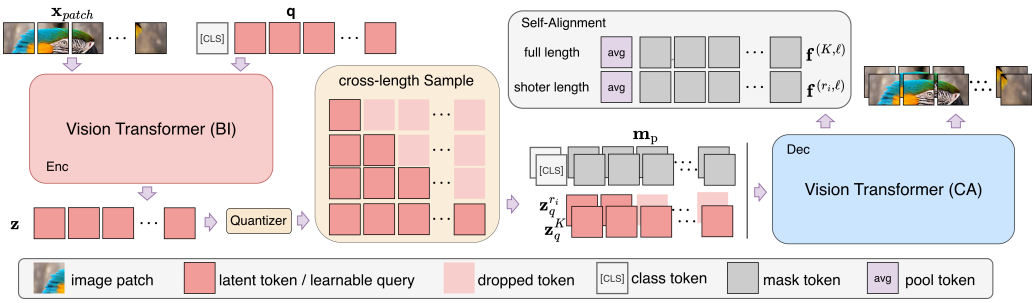


Figure 2: **Overview of the nested self-aligned training pipeline.** We train a 1D variable-length tokenizer using two components: (1) cross-length sampling, which jointly optimizes reconstruction from a full-length sequence and a sampled shorter sequence at each iteration; and (2) a nested self-alignment loss, which aligns latent representations across the two lengths to promote cross-length consistency and improve reconstruction.

latent sequence. At each training iteration i, we retain the full-length sequence $\mathbf{z}_q^{(K)}$ and sample a shorter length sequence $\mathbf{z}_q^{(r_i)}$, where $r_i \sim \{1, \ldots, K\}$. The resulting sequence $\mathbf{z}_q^{(r_i)}$ is therefore a prefix of $\mathbf{z}_q^{(K)}$. Both sequences are decoded, and their losses are accumulated to jointly optimize the tokenizer:

$$\mathcal{L}_{\text{joint}} = \mathcal{L}_{\text{rec}}(\mathbf{x}; K) + \mathcal{L}_{\text{rec}}(\mathbf{x}; r_i), \tag{2}$$

The two samples share all tokenizer parameters. The full-length sequence ensures that every latent token position is retained at least one decoding pass per iteration, while the shorter sequence trains the tokenizer to reconstruct images with a variable length to compress the reconstruction information.

Nested Self-Alignment Loss. Although cross-length joint training provides consistent supervision for full length reconstruction, it does not explicitly encourage feature consistency across token lengths. Empirically, we observe that shorter prefixes improve slowly and retain a substantial reconstruction gap relative to full-length sequences. Despite parameter sharing, improvements in full length reconstruction do not necessarily translate into comparable improvements at shorter lengths. This motivates an explicit mechanism for transferring alignment information at feature level across lengths. To this end, we introduce a nested self-alignment loss. With $n \in \{r_i, K\}$, we extract the output feature of the ℓ-th layer of the decoder:

$$\mathbf{h}_z^{(n,\ell)}, \mathbf{h}_{\text{p}}^{(n,\ell)} = \mathcal{D}_\phi^{(1:\ell)}(\mathbf{z}_q^{(n)}, \mathbf{m}_{\text{p}}), \tag{3}$$

where $\mathcal{D}_\phi^{(1:\ell)}$ denotes Parameters of the first ℓ layers of the decoder and $\mathbf{h}_z^{(n,\ell)} \in \mathbb{R}^{n \times d}$ and $\mathbf{h}_{\mathbf{p}}^{(n,\ell)} \in \mathbb{R}^{N \times d}$ denote the features of latent tokens and mask tokens, respectively. Unless otherwise specified, we apply alignment at the first decoder layer, i.e., $\ell = 1$.

Since the two sampled sequences contain different numbers of latent tokens, we apply average pooling along the token dimension to obtain a pooled latent feature, $\bar{\mathbf{h}}_z^{(n,\ell)} = \text{Avg}(\mathbf{h}_z^{(n,\ell)})$, where $\mathbf{h}_z^{(n,\ell)}$ contains the n latent-token features. We then concatenate this feature, treated as a single token, with the mask-token features along the token dimension to form $\mathbf{f}^{(n,\ell)} = \text{Concat}(\bar{\mathbf{h}}_z^{(n,\ell)}, \mathbf{h}_{\text{p}}^{(n,\ell)}) \in \mathbb{R}^{(N+1) \times d}$. We define the nested self-alignment loss as:

$$\mathcal{L}_{\text{align}} = 1 - \frac{1}{N+1} \sum_{j=1}^{N+1} \text{sim}(\mathbf{f}_j^{(r_i,\ell)}, \text{stopgrad}[\mathbf{f}_j^{(K,\ell)}]), \tag{4}$$

The full-length features serve as an online alignment target for shorter sequences. This supervision in feature space encourages shorter sequences to rapidly align their representations with those of the full length sequences, improving reconstruction quality at shorter token length without requiring an additional teacher network.

Overall objective of Nested Self-Aligned Learning. As shown in Fig. 2, our training framework combines cross-length joint training with nested self-alignment. Cross-length joint training decodes the full-length sequence at every iteration, so every position receives reconstruction supervision. Tail

tokens can therefore reduce the full-length loss only by adding details that the prefix lacks, which makes them residual refinements rather than redundant copies. Second, the self-alignment loss pulls the decoder features of each prefix toward those of the full-length sequence, which serves as a stop-gradient target. Since the prefix is a prefix of the full sequence, the most effective way to shrink this gap is to place the information in the earliest tokens that best approximates the full representation, such as global layout and semantics, and to leave what the prefix cannot capture to later tokens. The overall training objective is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{joint}} + \mathcal{L}_{\text{align}} \tag{5}$$

3.3 Variable-length Autoregressive Model

Variable-length AR generation relies on an ordered token sequence, in which every prefix is a progressively refined description of the same image. Nested dropout provides only a weak form of this ordering. Because early positions are retained under almost every truncation, they are pushed to carry the most important content. Later positions, however, are rarely supervised. As a result, the sequence tends to become *informative head, idle tail* rather than coarse-to-fine. NesTok strengthens the ordering in two complementary ways. For the downstream generator, this means that early predictions decide the global content, and later predictions refine it conditioned on that content. This ordering is consistent with next-token prediction.

Autoregressive Modeling. We model discrete visual token sequences through standard autoregressive next-token prediction:

$$p_\theta(\mathbf{z}_{1:n}) = \prod_{i=1}^{n} p_\theta(\mathbf{z}_i \mid \mathbf{z}_{<i}), \tag{6}$$

where θ denotes the model parameters and n is the token sequence length. We train the model on full-length sequences by minimizing the cross-entropy loss. At inference time, a desired token budget $n \leq K$ can be specified, where K is the maximum sequence length. The same generator then samples an n-token prefix, which is decoded into an image by the tokenizer decoder. Thus, a single trained generator supports variable-length inference, enabling flexible trade-offs between generation quality and computational cost. We additionally employ KV-cache to accelerate autoregressive sampling.

4 Experiment

4.1 Implementation Details

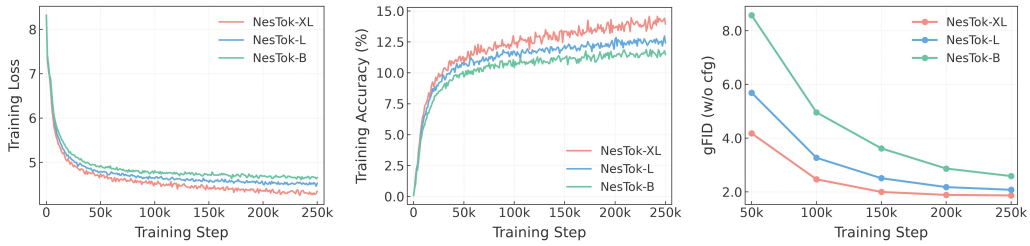
Model Configuration and Evaluation Our 1D variable-length tokenizer is based on One-D-Piece (Miwa et al., 2025) and uses ViT-B as the encoder and ViT-L as the decoder, with a codebook of 4,096 entries and a latent dimension of 32. Our autoregressive model adopts the decoder-only Transformer architecture of LlamaGen (Sun et al., 2024a). To accommodate 1D latent sequences, we replace 2D rotary positional embeddings (RoPE) with 1D RoPE (Su et al., 2024). We named our tokenizer as NesTok and denote the corresponding autoregressive models at different scales as NesTok-B/L/XL. Following prior work, we evaluate performance using Fréchet Inception Distance (FID), including reconstruction FID (rFID) and generation FID (gFID), as well as Inception Score (IS). We compute gFID on 50,000 generated images using the ADM's TensorFlow evaluation suite (Dhariwal & Nichol, 2021) and evaluate rFID using the MAR evaluation code (Li et al., 2024).

Training of Tokenizer. We train NesTok in three stages without any external teacher for distillation or representation alignment. The nested self-alignment loss is applied throughout training, with a batch size of 256. **In the first stage**, we jointly optimize the reconstruction, perceptual, and quantization losses for 200K iterations, warming up the learning rate to 1e-4. **In the second stage**, we disable the LPIPS of the perceptual loss, retaining only the loss computed using ConvNeXt-S. This stage lasts 200K iterations, during which the learning rate decreases from 5e-5 to 2e-5. **In the third stage**, we additionally introduce the GAN loss to improve the visual fidelity of reconstructions. We train for 150K steps, decreasing the learning rate from 2e-5 to 5e-6.

Training of Autoregressive Model. Following common practice, we train the LlamaGen with a batch size of 2,048 and a learning rate of 4e-4 with a warmup for 100 epochs. We train for 400 epochs totally, corresponding to approximately 250K steps. We apply QK-Norm (Team, 2024) in

Table 1: System-level comparison of different tokenizers and generation models on ImageNet 256×256. ↓ and ↑ indicate whether lower or higher values are better. We categorize tokenizers into three groups: 2D tokenizers, fixed-length 1D tokenizers, and variable-length 1D tokenizers.

Method	Tokenizer				Generator		w/o guidance		w/ guidance	
	Type	#Params	#Tokens	rFID↓	Type	#Params	gFID↓	IS↑	gFID↓	IS↑
2D Tokenization										
DiT-XL/2 (Peebles & Xie, 2023)	SD-VAE	84M	256	0.62	Diff.	675M	9.62	121.5	2.27	278.2
REPA-XL/2 (Yu et al., 2024d)	KL	84M	1024	0.62	Diff.	675M	5.90	157.8	1.42	305.7
Lightning-DiT-XL (Yao et al., 2025)	KL	84M	1024	0.28	Diff.	675M	2.17	205.6	1.35	295.3
MAR-L (Li et al., 2024)	KL	66M	256	0.87	MAR Diff.	479M	2.60	221.4	1.78	296.0
VQGAN (Esser et al., 2021)	VQ	23M	256	4.98	AR	1.4B	15.78	74.3	-	-
MaskGIT (Chang et al., 2022)	VQ	66M	256	2.28	Mask	227M	6.18	182.1	-	-
LlamaGen-XL (Sun et al., 2024a)	VQ	72M	256	0.94	AR	775M	14.77	80.8	2.62	244.1
RAR-L (Yu et al., 2025)	VQ	66M	256	2.28	AR	461M	5.39	149.1	1.70	299.5
IBQ-L (Shi et al., 2025)	IBQ	128M	256	1.37	AR	649M	-	-	2.45	267.5
VAR-d20 (Tian et al., 2024)	MSRQ	109M	680	0.90	VAR	600M	-	-	2.57	302.6
AliTok-L (Wu et al., 2025)	VQ	390M	273	0.86	AR	318M	1.98	200.8	1.38	326.2
1D Tokenization										
TiTok-L (Yu et al., 2024b)	VQ	641M	32	2.21	Mask	177M	3.15	173.0	2.77	199.8
GigaTok (Xiong et al., 2025)	VQ	622M	256	0.81	AR	111M	-	-	3.26	221.0
SoftVQ-VAE-L (Chen et al., 2025)	KL	176M	64	0.61	Diff.	675M	5.83	141.3	2.93	268.5
ResTok (Zhang et al., 2026)	VQ	662M	128	1.28	HAR	326M	-	-	2.34	257.8
MacTok (Zeng et al., 2026)	KL	675M	128	0.43	Diff.	326M	3.12	186.2	1.50	299.8
SemTok-XL (Qu et al., 2026)	VQ	2.35B	256	0.88	Mask	746M	-	-	2.54	305.6
SemTok-XXL	VQ	2.35B	256	0.88	Mask	1.2B	-	-	2.34	310.5
SpectralAR-d20 (Huang et al., 2025)	VQ	-	64	4.03	AR	600M	-	-	2.49	305.4
D-AR-L (Gao & Shou, 2026)	VQ	300M	256	1.52	AR	343M	-	-	2.44	262.9
D-AR-XL	VQ	300M	256	1.52	AR	775M	-	-	2.09	298.4
EOSTok-H (Chu et al., 2026)	IBQ	388M	256	0.71	AR	644M	1.48	239.5	1.38	265.7
1D Variable-length Tokenization										
FlexTok d18-d18-32 (Bachmann et al., 2025)	FSQ	950M	1-256	1.61	AR	1.33B	-	-	2.02	-
FlexTok d18-d28-32	FSQ	2.5B	1-256	1.45	AR	1.33B	-	-	1.86	-
Semanticist-XL-32 (Wen et al., 2025)	KL	-	1-256	**0.78**	AR Diff.	343M	-	-	2.57	254.0
One-D-Piece (Miwa et al., 2025)	VQ	641M	1-256	1.08	Mask	318M	-	-	2.35	224.4
DetailFlow-32 (Liu et al., 2025)	VQ	64M	1-256	0.80	AR	326M	-	-	2.75	250.8
ReTok (Fu et al., 2026)	VQ	232M	1-256	1.01	AR	775M	-	-	2.27	245.9
NesTok-B	VQ	390M	1-256	0.98	AR	177M	2.59	181.6	1.73	250.5
NesTok-L	VQ	390M	1-256	0.98	AR	318M	2.08	205.6	1.50	278.9
NesTok-XL	VQ	390M	1-256	0.98	AR	662M	**1.87**	**238.9**	**1.46**	**295.9**



(a) Training loss cuvers (b) Training accuracy cuvers (c) Traning gFID curves (w/o cfg)

Figure 3: **Training curves.** (a) Training loss. (b) Training accuracy (%). (c) gFID without classifier-free guidance across training steps and model sizes.

the attention modules and use RMSNorm (Zhang & Sennrich, 2019) for normalization. Before AR training, we precompute and cache the discrete visual token sequences using the tokenizer's encoder and quantizer to accelerate training.

4.2 Main Results

Generation Results on ImageNet 256×256. We evaluate NesTok against state-of-the-art methods on ImageNet-1K at 256 × 256 resolution. As shown in Tab. 1, NesTok achieves strong downstream autoregressive generation performance while maintaining high reconstruction quality. Specifically, NesTok achieves an rFID of 0.98, outperforming One-D-Piece's 1.08 with fewer tokenizer parameters (390M vs. 641M) and without external distillation. Among the compared 1D variable-length methods, NesTok-XL achieves the state-of-the-art gFID of **1.46**, compared with 2.35 for One-D-Piece. It achieves a gFID of **1.87** without classifier-free guidance, outperforming the other variable-length methods. Although EOSTok achieves the best generation quality among the compared 1D tokenization methods, it jointly trains the tokenizer and AR generator and relies on DINOv2-based distillation. Fig. 4 presents qualitative generation results obtained with NesTok. In Fig. 5, as the to-



Figure 4: **Examples of image generation with NesTok-XL on ImageNet** 256×256**.**

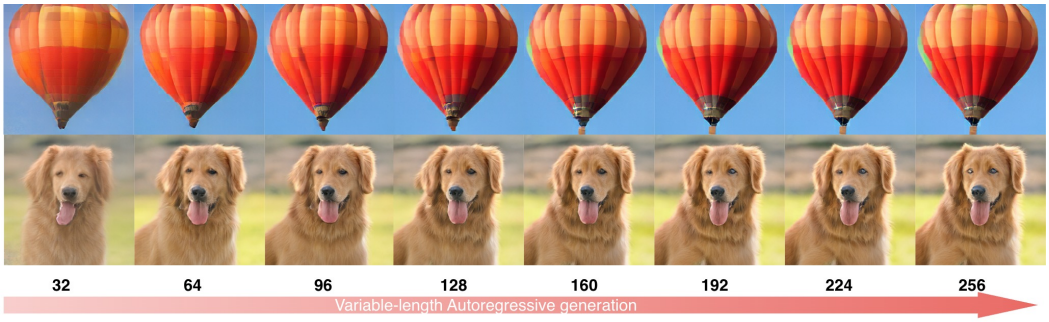


Figure 5: **Visualization of variable-length generation on ImageNet** 256×256 **resolution.** Images are generated using NesTok-XL with 32-256 tokens.

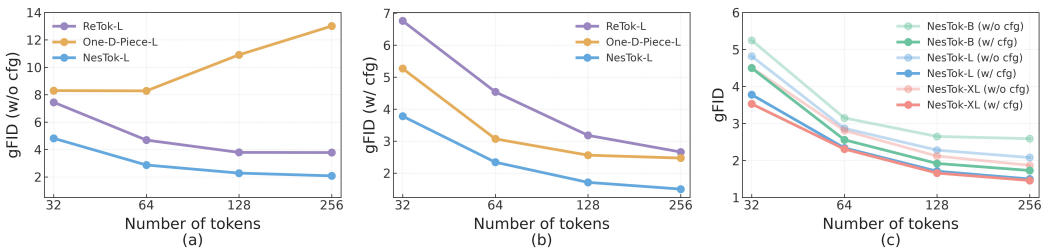


Figure 6: **gFID w/ and w/o cfg across different methods and model sizes.**

ken length increases from 32 to 256, the images exhibit progressively finer details while maintaining consistent global structure.

Variable-length Autoregressive Model. As shown in Fig. 6, for each model configuration, we evaluate a single trained AR generator across token lengths with and without classifier-free guidance (cfg). We compare NesTok with One-D-Piece-L (Miwa et al., 2025) and ReTok-L (Fu et al., 2026). NesTok-L substantially outperforms both methods across all evaluated token lengths. While One-D-Piece-L exhibits degraded generation quality at larger budgets without cfg, ReTok-L shows diminishing gains: increasing the sequence length from 128 to 256 tokens reduces gFID by only 0.01, from 3.79 to 3.78. Over the same interval, NesTok-L and NesTok-XL reduce gFID by 0.20 and 0.25, respectively. In Fig. 6 (c), NesTok consistently benefits from increasing the token lengths, with improvements continuing at longer sequence lengths. These results demonstrate the *autoregressive compatibility* of the learned representations and suggest that our nested self-alignment training architecture enables more effective use of additional tokens, including those at later positions.

Training Analysis. Fig. 3 illustrates the training dynamics and generation performance of autoregressive models built on NesTok. Fig. 3 (a) and (b) show steadily decreasing training loss and increasing training accuracy, respectively. Fig. 3 (c) reports gFID scores using 256 tokens without classifier-free guidance. Scaling up the AR generator consistently improves gFID. These results support the *autoregressive compatibility* of the token sequences learned by NesTok, suggesting that our training strategy produces representations well suited to autoregressive prediction.

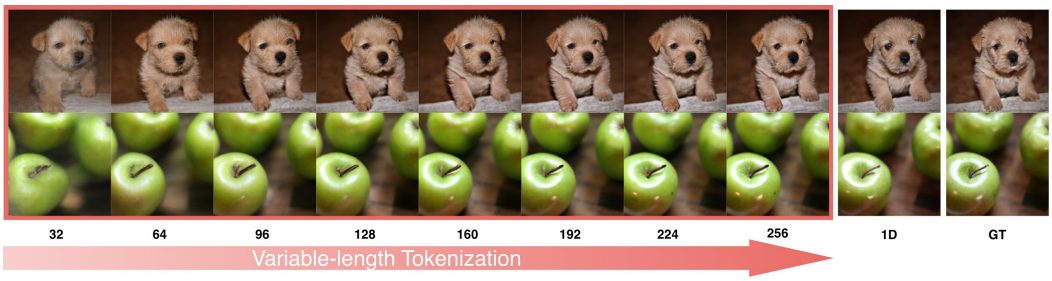


Figure 7: **Visualization of variable-length reconstruction on ImageNet 256×256 resolution.** 1D denotes the 1D Tokenizer with 256 tokens.

Table 2: **Ablation study of different training settings.** We report generation results using LlamaGen-Base (177M) trained for 200 epochs.

Training Setting	AR Training		Evaluation							
	Loss↓	Acc.↑	rFID↓				gFID↓			
			32	64	128	256	32	64	128	256
1D Tokenizer	5.37	7.9%	-	-	-	0.83	-	-	-	2.33
+Nested Dropout	2.37	46.2%	4.28	2.88	2.07	1.93	5.98	4.64	3.12	2.46
+Cross-length Sample	4.70	11.3%	3.70	2.14	1.43	0.98	5.76	4.16	2.36	1.93
+Nested Self-Alignment Loss	4.68	11.4%	**3.49**	**2.10**	**1.39**	**0.98**	**5.48**	**3.85**	**2.18**	**1.92**

4.3 Ablation Studies.

Ablation of Components. We conduct an ablation study to evaluate the contributions of individual components. As shown in Table 2, the fixed-length 1D Tokenizer achieves the best rFID but does not support variable-length representations. Introducing nested dropout enables variable-length tokenization but degrades reconstruction quality. This variant achieves an AR training accuracy of 46.2%, yet yields a gFID of 2.46. According to Fig. 1, the result suggests that nested dropout is associated with concentrated code usage and tail-token collapse. Cross-length joint training mitigates this issue, ensuring that all latent-token positions participate in reconstruction training. Adding the nested self-alignment loss further promotes *cross-length consistency* and improves reconstruction from shorter sequences. Although this loss yields little improvement in full-length generation quality, it substantially improves generation at shorter lengths. With both components, NesTok achieves a gFID of 1.92, outperforming the 1D Tokenizer's 2.33. Fig. 7 further shows that NesTok preserves the main visual content with short prefixes and restores finer details as the token length increases.

Comparison of sampling speed. We compare the sampling speed of different methods in terms of throughput. Although One-D-Piece achieves high throughput with its MaskGIT generator, its generation quality is comparatively limited. Semanticist employs a diffusion decoder and achieves a throughput of only 1.03 images per second even with 32 tokens, limiting its suitability for applications requiring rapid responses. NesTok achieves the best gFID among the compared methods while maintaining competitive throughput, offering a favorable trade-off between generation quality and sampling efficiency.

Table 3: **Comparison of sampling speed.** Throughput (images/s) measured on a single H800 GPU with a batch size of 128, averaged over ten sampling runs using the official implementation of each method.

Method	Tokens	Params	gFID	images/s
ReTok-L	256	343M	2.27	10.11
One-D-Piece-L	256	318M	2.35	42.67
DetailFlow	256	326M	2.75	19.69
Semanticist-L	32	343M	2.57	1.03
NesTok-L	256	318M	**1.50**	16.35

5 Conclusion

We presented NesTok, a variable-length 1D tokenizer trained through nested self-alignment without external distillation. By combining cross-length joint training with feature alignment, our framework mitigates tail-token collapse, increases average codebook utilization to nearly 100%. The learned representations support coarse-to-fine refinement and exhibit improved autoregressive compatibil-

ity. The AR generator trained with NesTok supports multiple token lengths, with generation quality improving consistently as more tokens are generated. The result demonstrates that an effective tokenizer training strategy can improve the variable-length image generation while retaining standard next-token prediction.

REFERENCES

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. Flextok: Resampling images into 1d token sequences of flexible length. In *Forty-second International Conference on Machine Learning*, 2025.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11305–11315. IEEE, 2022.

Hao Chen, Ze Wang, Xiang Li, Ximeng Sun, Fangyi Chen, Jiang Liu, Jindong Wang, Bhiksha Raj, Zicheng Liu, and Emad Barsoum. Softvq-vae: Efficient 1-dimensional continuous tokenizer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28358–28370. IEEE, 2025.

Yitong Chen, Zuxuan Wu, Xipeng Qiu, and Yu-Gang Jiang. Catok: Taming mean flows for one-dimensional causal image tokenization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23161–23171, June 2026.

Wenda Chu, Bingliang Zhang, Jiaqi Han, Yizhuo Li, Linjie Yang, Yisong Yue, and Qiushan Guo. End-to-end autoregressive image generation with 1d semantic tokenizer. *arXiv preprint arXiv:2605.00503*, 2026.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Shivam Duggal, Phillip Isola, Antonio Torralba, and William T Freeman. Adaptive length image tokenization via recurrent allocation. In *First Workshop on Scalable Optimization for Efficient and Adaptive Foundation Models*, 2025.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Zixuan Fu, Lanqing Guo, Chong Wang, Binbin Song, Ding Liu, and Bihan Wen. Improving flexible image tokenizers for autoregressive image generation. *arXiv preprint arXiv:2601.01535*, 2026.

Ziteng Gao and Mike Zheng Shou. D-ar: Diffusion via autoregressive models. In *International Conference on Learning Representations*, volume 2026, pp. 68013–68032, 2026.

Qiyuan He, Yicong Li, Haotian Ye, Jinghao Wang, Xinyao Liao, Pheng-Ann Heng, Stefano Ermon, James Zou, and Angela Yao. Rear: Rethinking visual autoregressive models via generator-tokenizer consistency regularization. *arXiv preprint arXiv:2510.04450*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Yuanhui Huang, Weiliang Chen, Wenzhao Zheng, Yueqi Duan, Jie Zhou, and Jiwen Lu. Spectralar: Spectral autoregressive visual generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15842–15852. IEEE, 2025.

Dongwon Kim, Ju He, Qihang Yu, Chenglin Yang, Xiaohui Shen, Suha Kwak, and Liang-Chieh Chen. Democratizing text-to-image masked generative models with compact text-aware one-dimensional tokens. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18442–18452. IEEE, 2025.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11513–11522. IEEE, 2022a.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11513–11522. IEEE, 2022b.

Changlin Li, Jiawei Zhang, Sihao Lin, Zongxin Yang, Junwei Liang, Xiaodan Liang, and Xiaojun Chang. Efficient training of large vision models via advanced automated progressive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(8):8798–8812, 2026. doi: 10.1109/TPAMI.2026.3673336.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37: 56424–56445, 2024.

Yiheng Liu, Liao Qu, Huichao Zhang, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Xian Li, Shuai Wang, Daniel K Du, et al. Detailflow: 1d coarse-to-fine autoregressive image generation via next-detail prediction. *arXiv preprint arXiv:2505.21473*, 2025.

Jiasen Lu, Liangchen Song, Mingze Xu, Byeongjoo Ahn, Yanjun Wang, Chen Chen, Afshin Dehghan, and Yinfei Yang. Atoken: A unified tokenizer for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28701–28711, June 2026.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. In *International Conference on Learning Representations*, volume 2024, pp. 51772–51783, 2024.

Keita Miwa, Kento Sasaki, Hidehisa Arai, Tsubasa Takahashi, and Yu Yamaguchi. One-d-piece: Image tokenizer meets quality-controllable compression. *arXiv preprint arXiv:2501.10064*, 2025.

Ziqi Pang, Tianyuan Zhang, Fujun Luan, Yunze Man, Hao Tan, Kai Zhang, William T Freeman, and Yu-Xiong Wang. Randar: Decoder-only autoregressive visual generation in random orders. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 45–55. IEEE, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Yunpeng Qu, Kaidong Zhang, Yukang Ding, Ying Chen, and Jian Wang. Semantic one-dimensional tokenizer for image reconstruction and generation. *arXiv preprint arXiv:2603.16373*, 2026.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Fengyuan Shi, Zhuoyan Luo, Yixiao Ge, Yujiu Yang, Ying Shan, and Limin Wang. Scalable image tokenization with index backpropagation quantization. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16037–16046. IEEE, 2025.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024a.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024b.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

XuDong Wang, Xingyi Zhou, Alireza Fathi, Trevor Darrell, and Cordelia Schmid. Visual lexicon: Rich image features in language space. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19736–19747. IEEE, 2025.

Xin Wen, Bingchen Zhao, Ismail Elezi, Jiankang Deng, and Xiaojuan Qi. "principal components" enable a new language of images. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16641–16651. IEEE, 2025.

Pingyu Wu, Kai Zhu, Yu Liu, Longxiang Tang, Jian Yang, Yansong Peng, Wei Zhai, Yang Cao, and Zheng-Jun Zha. Alitok: Towards sequence modeling alignment between tokenizer and autoregressive model. *arXiv e-prints*, pp. arXiv–2506, 2025.

Tianwei Xiong, Jun Hao Liew, Zilong Huang, Jiashi Feng, and Xihui Liu. Gigatok: Scaling visual tokenizers to 3 billion parameters for autoregressive image generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18770–18780. IEEE, 2025.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15703–15712. IEEE, 2025.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10459–10469. IEEE, 2023.

Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion-tokenizer is key to visual generation. In *International Conference on Learning Representations*, volume 2024, pp. 765–783, 2024a.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024b.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024c.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18431–18441. IEEE, 2025.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024d.

Hengyu Zeng, Xin Gao, Guanghao Li, Yuxiang Yan, Jiaoyang Ruan, Junpeng Ma, Haoyu Albert Wang, and Jian Pu. Mactok: Robust continuous tokenization for image generation. *arXiv preprint arXiv:2603.29634*, 2026.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in neural information processing systems*, 32, 2019.

Xu Zhang, Cheng Da, Huan Yang, Kun Gai, Ming Lu, and Zhan Ma. Restok: Learning hierarchical residuals in 1d visual tokenizers for autoregressive image generation. *arXiv preprint arXiv:2601.03955*, 2026.

Chuanxia Zheng and Andrea Vedaldi. Online clustered codebook. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22741–22750. IEEE, 2023.