Balancing Image Compression and Generation with Bootstrapped Tokenization

Haozhe Chi [1] **Jinghan Li** [1] **Hao Jiang** [1] **Wu Sheng** [1] **Yi Ma** [2] **Jing Wang** [2] **Yadong Mu** [1*]
[1]Peking University, [2]Central Media Technology Institute, Huawei

Abstract

Despite progress in image tokenization, standard methods encode redundant information by mixing all granularities within each token, thus redundancy persists between tokens. The mix of information of different granularity also complicates the training of generators. This paper introduces SelfBootTok, a method that resolves this by cleanly decomposing information into global and local token groups. Through self-bootstrapped learning, the model predicts local details exclusively from global tokens, shifting the burden of visual details from the generator to the tokenizer. Consequently, our generator is far more efficient, requiring only global tokens and reducing computation by approximately 40%, while delivering superior reconstruction and generation. Moreover, this paradigm scales elegantly: by leveraging more data or parameters to self-supervise local representation learning, SelfBootTok achieves a new state-of-the-art gFID score of 1.56 using only 64 tokens.

1 Introduction

Recent advancements in diffusion-based image generation, exemplified by the Diffusion Transformer (DiT) [46] and flow matching [39, 21, 15], have driven significant progress. Beyond diffusion models, other generative paradigms like masked generative models and autoregressive visual models have seen notable developments. For instance, VAR [51] introduces next-scale prediction, where autoregressive visual models surpass diffusion models. Additionally, MAE-tok [11] and MAR [36] have improved masked generative modeling, boosting downstream generation performance. Multi-scale designs, as demonstrated by HieraTok [10], VAR [51], and FlowAR [47], further enhance performance. In parallel with quality improvements, recent studies have focused on enhancing generation efficiency. For example, Lightning-DiT [62] accelerates training by optimizing latent representations for image generation, addressing the reconstruction-generation tradeoff. Similarly, REG [59] accelerates SiT [42] training. As generative frameworks advance, image tokenization has become essential for enabling multimodal understanding and generation through compact latent representations. Early methods like VQ-GAN [18] and VQ-VAE [54] encoded images as 2D grid latents, preserving spatial relationships. However, the need for a strict one-to-one mapping between image patches and tokens is unclear, driving the exploration of 1D sequential tokenizers, which offer higher compression while retaining key semantic and structural information.

The emergence of 1D image tokenizers to encode images into compact latent sequences provides a promising alternative to conventional 2D grid representations. The key challenge lies in capturing both global semantics and local details within an extremely limited number of tokens. Early efforts such as Titok [51] demonstrated that an image can be effectively represented by as few as 32 one-dimensional tokens, establishing the feasibility of this paradigm. Subsequent developments, including Flextok [5], TA-Titok [28], Flowtok [24], and GigaTok [61], have progressively enhanced token efficiency, flexibility, and semantic richness. Despite these advancements, existing methods still

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

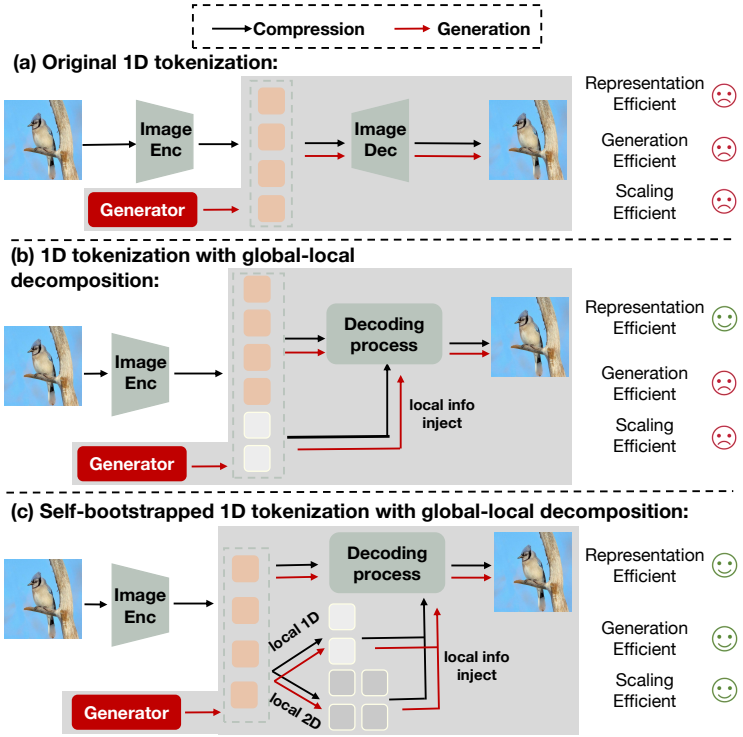

Figure 1: Illustration of our self-bootstrapped learning paradigm. Compared to classical 1D image tokenizers [65, 12, 61] and recent approaches incorporating local detail injection [10, 19], our method employs a global-local decomposition to achieve efficient hierarchical representation learning and adopts a self-bootstrapped strategy to enable both efficient generation and scalable training.

exhibit redundant token interactions and overlap between global and local representations, which constrain scalability and generation efficiency.

In this work, we propose SelfBootTok. It improves the balance at various aspects between image compression and generation, by decomposing conventional visual tokens into high-level semantics-based (, global tokens) and fine-grained visual information-related (, local tokens). The conceptual comparison with existing works is illustrated in Fig. 1. The proposed new model has several advantages:

1. Simpler and more efficient generator: Traditionally, visual generators are required to produce all levels of visual details at once when fed with prompts. In contrast, our method bootstraps fine-grained visual information directly from global tokens using unlabeled images, as shown in Fig. 1, thereby bypassing the need for extensive text-image pairs. This scheme essentially moves part of the generation process into the compression pipeline. The generator can focus solely on learning high-level semantics from a compact token set, requiring significantly less data and complexity.

2. Scalable image tokenizer: Our tokenizer employs a self-supervised learning strategy to reconstruct local image information from global tokens. It utilizes a hybrid of 1D and 2D local tokens to capture features at varying granularities. A novel optimal transport alignment is introduced to compactly map these 2D features into the 1D token sequence. This paradigm allows the tokenizer to scale efficiently using more data or parameters. This decomposition also minimizes token-level redundancy, thereby reducing the computational burden on the subsequent generator and enhancing the generation efficiency.

3. Parallel optimization of scaled-up tokenizers and generators: This framework enables efficient scaling by sharing only a set of global tokens. After learning these tokens, the global component of the tokenizer is frozen. The generators and the larger, bootstrapped local components of the tokenizers can then be trained in parallel. Crucially, this decouples their design, allowing the tokenizer size to be varied without requiring the generators to be re-trained.

Comprehensive experiments demonstrate that our method achieves state-of-the-art generation performance among 1D tokenizers and exhibits strong scalability of the self-bootstrapping design.

Moreover, we propose a training strategy that scales local aligners while generating global tokens only once, reducing total computational cost by approximately 40% and training time by about 54%.

2 Related work

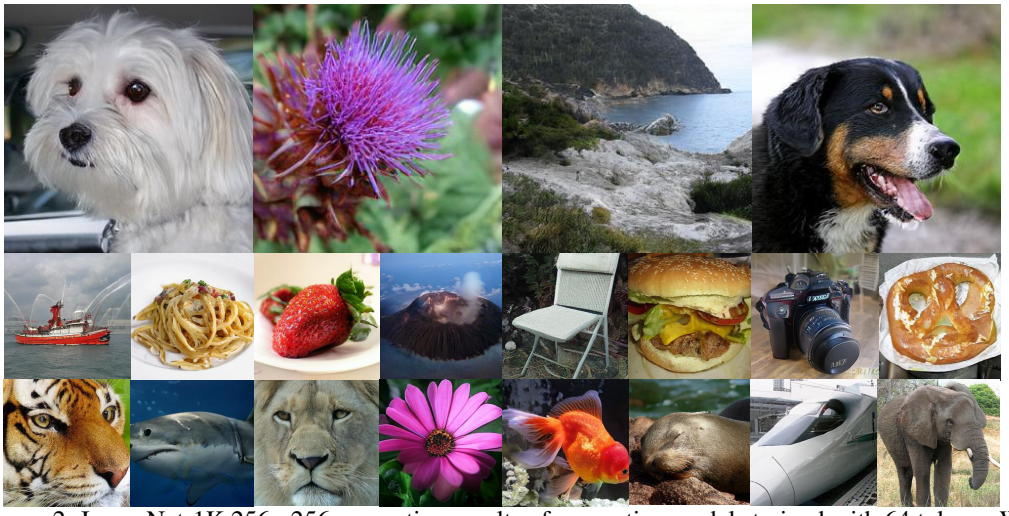

Figure 2: ImageNet-1K 256×256 generation results of generative models trained with 64 tokens. We include a versatile classes of images such as animals, plants and food. Our method achieve pleasant generation outcomes with efficient token representation and local details.

2.1 Image tokenization

Image tokenization methods can be broadly categorized by their latent space type (discrete vs. continuous) and spatial structure (2D grid vs. 1D sequence). For discrete modeling, Vector Quantization (VQ) frameworks [54, 18, 63] learn a codebook of discrete visual tokens. In contrast, continuous models such as KL-VAEs [29] employ the reparameterization trick to constrain latents to Gaussian priors. Most early works, including VQ-VAE and SD-VAE [48], adopt 2D grid latents to preserve spatial structure, a design that underpins diffusion-based models like Stable Diffusion. While highly effective, these 2D formulations impose limitations in compactness and efficiency, prompting recent research into 1D image tokenizers that represent images as highly compressed sequential embeddings.

The study of 1D image tokenization began with Titok [65], which demonstrated effective reconstruction using only 32 tokens by representing images as short 1D sequences. Titok uses a two-stage training pipeline for discrete 1D VQ modeling, leveraging codes from pretrained models like MaskGIT-VQGAN [9]. Subsequent works have explored various directions, including Flowtok and TA-Titok [24, 28] for text-to-image generation, Flextok and OneDpiece [5, 44] for variable-length sequences, and GigaTok [61] for combining 1D and 2D structures. Recent work [7] further investigates 1D sequences for fine-grained image editing. To improve tokenization quality, SoftVQ [12] introduces a differentiable soft vector quantization mechanism, and MAE-Tok [11] leverages masked autoencoding for semantically enriched latent spaces with strong reconstruction fidelity. However, none of these tokenizers explicitly differentiate or exploit the global-local information. To address this, we propose SelfBootTok, a 1D tokenizer that models global-local interactions for compact, efficient image representation.

2.2 Generative Models

Generative models are critical for evaluating image tokenizers in downstream generation tasks, and can be broadly divided into diffusion-based and autoregressive approaches. Diffusion-based models generate images by gradually converting Gaussian noise into structured latent codes. Representative methods include DiT, SiT, and MAR [46, 42, 36]. DiT adopts a Transformer backbone and models diffusion via stochastic differential equations. SiT leverages stochastic interpolants and optimizes the velocity field of a probability flow ODE. MAR unifies diffusion and autoregression within an encoder-decoder Transformer. Lightning-DiT [62] further boosts efficiency via a lightweight design

that balances reconstruction and generation quality. Autoregressive models follow the next-token prediction paradigm from LLMs, enabling unified multimodal generation. Methods incorporating masked modeling [9, 58, 64, 65] and next-scale prediction [51, 37] further improve representation and generation quality.

2.3 Self-Bootstrapped and Scaling Paradigms in Multimodal Learning

Self-bootstrapping has proven effective for multimodal learning by fully exploiting model capacity. For instance, BLIP-2 [33] introduced it for vision–language fusion, OK-VQA [23] and VILA [20] applied it to visual question answering and model refinement, and VideoJudge [55] extended it to scalable video assessment. Further works [60, 17] validated its potential in data-efficient reasoning and 3D scene understanding. Nevertheless, its application to image tokenization remains largely unexplored. The success of scaling in LLMs has motivated extensive research in multimodal understanding and generation. In visual understanding, recent studies have focused on scaling vision encoders within multimodal LLMs [2, 4, 14, 34, 38, 43, 40, 1, 50, 13, 67, 68], with InternVideo2 [57] and CuMo [32] adopting efficient scaling and sparsely-gated Mixture-of-Experts layers. For image understanding, SViT [69], LongLLaVA [56], and LLaVA-scale [41] investigate backbone scaling, while AuroraCap [8] and LLaVA-Next [31] use multi-stage training to unify visual instruction tuning across image and video. In multimodal generation, scaling remains a critical challenge. Methods such as ViT-VQGAN [63] and ViTok [22] show that larger tokenizers do not always improve generation performance and often underuse model capacity. GigaTok [61] improves scaling efficiency via hybrid CNN-Transformer tokenizers. However, efficiently scaling local token information is still an open problem, which we target in this work.

3 Method

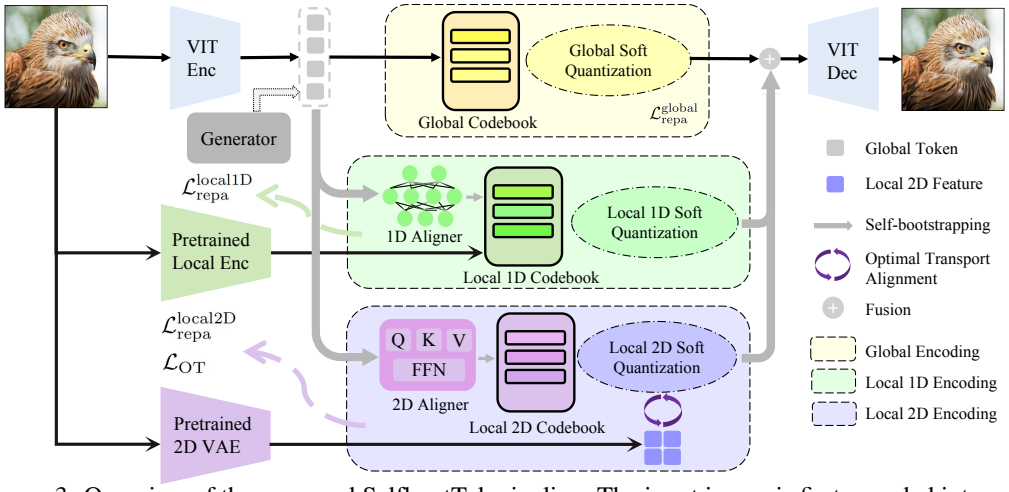

Figure 3: Overview of the proposed SelfbootTok pipeline. The input image is first encoded into a set of global tokens using a ViT backbone. Subsequently, local tokens of varying granularity (i.e., 1D or 2D) are predicted through a self-bootstrapping paradigm. These local tokens are aligned with different pretrained visual encoders to capture multi-granularity structural information. Both global and local tokens are then softly quantized, fused, and finally decoded using a ViT decoder. This overall architecture offers strong scalability and training efficiency, enabling high-quality reconstruction and generation with minimal computational overhead.

Our image tokenization framework consists of three key components: global-local decomposition, self-supervised bootstrapped learning, and 2D to 1D alignment via optimal transport. **Our framework employs a hierarchical self-bootstrapped VAE, which theoretically achieves superior performance (see Appendix Section A for detailed proofs).** The global-local decomposition aims to separately optimize global and local token representations, thereby reducing redundant information and improving representational efficiency. The self-supervised bootstrapped learning mechanism is designed to capture the intrinsic relationships between global semantics and fine-grained local details, facilitating hierarchical feature learning. The 2D to 1D alignment module preserves the sequential

structure of 1D tokens by transporting the 2D token distribution into a 1D sequence space and minimizing an optimal transport loss. Moreover, our framework naturally supports efficient scaling of local aligners, enabling flexible adaptation to different computational budgets while maintaining high generation performance and significantly reducing training cost.

3.1 Global-local Decomposition

As shown in Fig. 3, after getting the global tokens z_G from input image I through a Vision Transformer encoder Enc, our pipeline recovers more local information through local alignment and get local 1D tokens z_{L1} and local 2D tokens z_{L2}. Then, we apply a multi-codebooks design that distinguishes local entries with global entries and enhances the localized representations with either 1D or 2D information injection:

$$\begin{aligned} &\textbf{Encode:} \quad \mathbf{z}_G = Enc(I), \quad \mathbf{z}_{\text{tot}} = \{\mathbf{z}_G, \mathbf{z}_{L1}, \mathbf{z}_{L2}\}, \\ &\textbf{Codebook Set:} \quad \mathcal{C}_{\text{tot}} = \{\mathcal{C}_{\text{global}}, \mathcal{C}_{\text{local1D}}, \mathcal{C}_{\text{local2D}}\}. \end{aligned} \tag{1}$$

We apply soft vector quantization [12] such that both global and local quantizations are fully-differentiable, and thus the encoder and codebook can be optimized directly from the reconstruction loss:

$$q_\phi(Z \mid I) = \text{Softmax}\Big(-\tfrac{1}{\tau}\|\hat{Z} - \mathcal{C}\|_2\Big), \ \hat{Z} \in \mathbf{z}_{\text{tot}}, \tag{2}$$

$$Z = q_\phi(Z \mid I)\,\mathcal{C}, \ \mathcal{C} \in \mathcal{C}_{\text{tot}}. \tag{3}$$

Under this design, we separate different token groups using distinct codebooks, allowing each group to specialize in learning representations of a specific granularity. The overall training objective combines reconstruction, perceptual, adversarial, alignment, and KL regularization terms:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_1 \mathcal{L}_{\text{percep}} + \lambda_2 \mathcal{L}_{\text{adv}} + \lambda_3 \mathcal{L}_{\text{align}} + \lambda_4 \mathcal{L}_{\text{KL}}, \tag{4}$$

The alignment loss $\mathcal{L}_{\text{align}}$ further decomposes into three components, corresponding to the global, local, and cross-dimensional (2D to 1D) alignment objectives:

$$\mathcal{L}_{\text{align}} = \mu_1 \mathcal{L}_{\text{repa}}^{\text{global}} + \mu_2 \mathcal{L}_{\text{repa}}^{\text{local}} + \mu_3 \mathcal{L}_{\text{OT}} \tag{5}$$

The global alignment loss encourages global tokens to align with the features extracted by a pretrained global semantic encoder (e.g., DINOv2 [45]). This loss is computed as a patch-wise similarity measure:

$$\mathcal{L}_{\text{repa}}^{\text{global}} = \text{PatchwiseSim}(\mathbf{z}_G, f_{\text{DINOv2}}(I)). \tag{6}$$

Similarly, the local alignment loss matches each local 1D token to its corresponding localized representation from pretrained encoders (e.g., SigLIP2 [52] or I-JEPA [3]):

$$\mathcal{L}_{\text{repa}}^{\text{local}} = \text{PatchwiseSim}(\mathbf{z}_{L1} \text{ or } \mathbf{z}_{L2}, f_{\text{local}}(I)). \tag{7}$$

Finally, $\mathcal{L}_{\text{OT}}$ denotes the optimal transport loss that aligns the 2D VAE latent representations of images with their corresponding 1D token sequences, as defined in Eq. (13). This design effectively disentangles global and local information across token groups, leading to more compact and efficient representations, particularly for global tokens.

3.2 Self-supervised Bootstrapped Learning

After obtaining the global tokens $\mathbf{z}_G$ from Eq. (1), we employ a self-bootstrapped paradigm to predict local tokens from these global representations. Specifically, to predict local 1D tokens $\mathbf{z}_{L1}$, we use an MLP module to project $\mathbf{z}_G$ into a higher-dimensional token space:

$$\mathbf{z}_{L1} = \text{MLP}(\mathbf{z}_G), \tag{8}$$

To predict local 2D tokens, we employ several scalable causal transformer blocks that autoregressively map $\mathbf{z}_G$ to another sequence of tokens $\mathbf{z}_{L2}$:

$$p_\theta(\mathbf{z}_{L2} \mid \mathbf{z}_G) = \prod_{i=1}^{N_l} p_\theta\big(\mathbf{t}_i^{\text{local}} \mid \mathbf{z}_G, \mathbf{t}_{<i}^{\text{local}}\big), \tag{9}$$

We then align this token sequence with the corresponding sequence transported from the 2D VAE latents, as described in Eq. (13). During the generation process, the generator produces only the global tokens $\mathbf{z}_G$. The local tokens $\mathbf{z}_{L1}$ and $\mathbf{z}_{L2}$ are then automatically predicted through the local 1D and 2D aligners. Finally, after the soft quantization of $\mathbf{z}_G$, $\mathbf{z}_{L1}$, and $\mathbf{z}_{L2}$, all tokens are fused and decoded together:

$$\hat{\mathbf{x}} = \mathrm{Dec}\big(\mathrm{Fuse}(\mathbf{z}_G, \mathbf{z}_{L1}, \mathbf{z}_{L2}); \mathbf{z}_{L1}; \mathbf{z}_{L2}\big), \tag{10}$$

The fused tokens, along with $\mathbf{z}_{L1}$ and $\mathbf{z}_{L2}$, are then injected into the cross-attention layers of the decoder. We explore two fusion strategies for decoding: soft residual fusion and concatenation fusion. Further ablation studies comparing these strategies are provided in the experimental section.

$$\mathrm{Fuse} = \begin{cases} \alpha\,\mathbf{z}_G + \beta\,\mathbf{z}_{L1} + (1-\alpha-\beta)\,\mathbf{z}_{L2}, & \text{(Soft residual)} \\ \mathrm{Concat}[\mathbf{z}_G; \mathbf{z}_{L1}; \mathbf{z}_{L2}], & \text{(Concatenation)} \end{cases} \tag{11}$$

Within our self-bootstrapped paradigm, generating only high-level, compact global information is sufficient to achieve high-quality generation. This approach reduces computational costs and optimizes the latent space of global tokens.

3.3 2D to 1D alignment via optimal transport

To efficiently inject 2D priors into a 1D token sequence and achieve a compact 1D representation, we use a local 2D aligner to predict a 1D sequence and align the 2D VAE latent features with it. **This formulation naturally fits into the optimal transport (OT) framework, which constructs mappings between tensors of different dimensions while encouraging compact representations, and can be efficiently solved via the Sinkhorn algorithm.** Specifically, let $\mathbf{X} \in \mathbb{R}^{B\times N_x\times d}$ be the matrix of 2D VAE latents (with $N_x = H \times W$ spatial tokens), and $\mathbf{z}_{L2} \in \mathbb{R}^{B\times N_z\times d}$ be the matrix of 1D tokens (with N_z latent tokens). Here, B denotes the batch size, and d refers to the token dimension. The goal is to find a transport plan $\mathbf{P} \in \mathbb{R}_+^{N_x\times N_z}$ that couples the marginals $\mathbf{a} \in \Delta^{N_x}$ and $\mathbf{b} \in \Delta^{N_z}$ (where Δ^N denotes the N-simplex).

Let the squared-Euclidean cost matrix be

$$\mathbf{C}_{ij} = \|\mathbf{x}_i - \mathbf{z}_j\|_2^2, \quad \text{for } i \in [1, N_x],\ j \in [1, N_z]$$

measuring pairwise distances between 2D latents $\mathbf{X}$ and 1D tokens $\mathbf{z}_{L2}$. We solve the entropy-regularized optimal transport problem

$$\mathbf{P}^* = \arg\min_{\mathbf{P}\in\Pi(\mathbf{a},\mathbf{b})} \langle \mathbf{P}, \mathbf{C}\rangle \ + \ \varepsilon\,\mathcal{H}(\mathbf{P}), \tag{12}$$

where $\Pi(\mathbf{a}, \mathbf{b})$ is the set of transport plans coupling $\mathbf{a}$ and $\mathbf{b}$, $\langle ., . \rangle$ represents the Frobenius inner product and $\mathcal{H}(\mathbf{P})$ is the entropy regularizer. The optimal transport plan $\mathbf{P}^*$ is computed using the Sinkhorn algorithm:

$$\mathbf{P}^* = \mathrm{Sinkhorn}(\mathbf{K}, \mathbf{a}, \mathbf{b}),$$

where $\mathbf{K} = \exp(-\mathbf{C}/\varepsilon)$. And we use the transport cost as part of the alignment loss shown in Eq. (5):

$$\mathcal{L}_{\mathrm{OT}} \ = \ \langle \mathbf{P}^*, \mathbf{C}\rangle. \tag{13}$$

3.4 Local Scaling with heavier local aligner

To explore the potential of the self-bootstrapped paradigm, we conduct experiments using local aligners at different spatial scales. Specifically, we vary the size of the local 2D module while keeping all other components fixed. When the local 2D aligner becomes large, the overall model becomes computationally heavy and difficult to optimize. To address this issue, we adopt a two-stage training strategy. In the first stage, we train the entire pipeline with a smaller local 2D aligner. In the second stage, we scale up the local 2D aligner and fine-tune it while freezing the remaining components. This strategy allows simultaneous training of the generator and the enlarged local aligner, enabling the generator to be trained once for global tokens and subsequently reused across different local aligner scales. Further discussions on the scaling results are provided in Section 4.2 and Fig. 6.

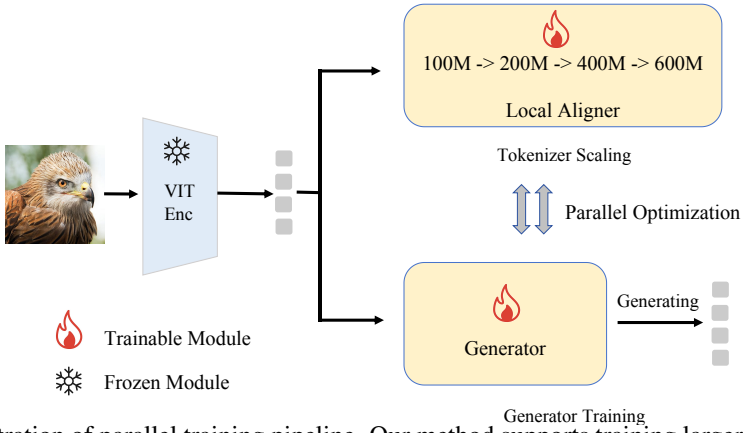

Figure 4: Illustration of parallel training pipeline. Our method supports training larger scale tokenizer and generator at the same time, which enhances training efficiency.

Tokenizer	#Tokens	Generators	rFID↓	gFID(w/o CFG)↓	gFID(w CFG)↓
2D Tokenizer					
VAE [48]	4096	LDM-4 [48]	0.27	10.56	3.60
SD-VAE [48]	256	DiT-XL/2 [46]	0.62	9.62	2.27
SD-VAE [48]	256	SiT-XL/2 [42]	0.62	8.61	2.06
VAR-Tok [51]	680	VAR-d30 [51]	1.00	–	1.92
VQGAN [18]	256	LlamaGen [49]	0.59	9.38	2.18
VA-VAE [62]	256	LightningDiT [62]	0.28	2.17	1.35
Auto-regressive Tokenizer					
MaskGIT [9]	256	MaskGIT [9]	2.28	6.18	–
LlamaGen-3B [49]	576	LlamaGen [49]	2.19	–	2.18
RQ-VAE [30]	256	RQ-Transformer [30]	3.20	–	3.80
MAGE [35]	256	MAGE [35]	–	6.93	–
MAR-H [36]	256	MAR-H [36]	1.22	2.35	1.55
ViTok S-B/16 [22]	256	DiT-XL [46]	–	–	2.45
1D Tokenizer					
TiTok-B [65]	64	MaskGIT-ViT [9]	1.70	–	2.48
TiTok-S [65]	128	MaskGIT-UViT-L [9, 6]	1.71	–	1.97
FlexTok d18-d28 [5]	1 to 256	LlamaGen [49]	1.45	–	1.86
OneD-piece [44]	1 to 256	–	1.08	–	–
HieraTok [10]	256	DiT [46]	0.45	3.53	1.82
GigaTok XL-XXL [61]	256	LlamaGen XXL [49]	0.79	–	1.98
MAETok [11]	128	LightningDiT [62]	0.48	–	1.73
SoftVQ [12]	64	SiT [42]	0.88	5.98	1.78
SelfBootTok	**64**	SiT [42]	0.66	2.22	1.56

Table 1: **Performance of SelfBootTok** in the context of state-of-the-art generative models on ImageNet-256. We additionally report the number of latent tokens (#Tokens) for each tokenizer. Our method achieves state-of-the-art generation performance gfid 1.56 among 1D tokenizers with only 64 tokens.

4 Experiments

4.1 Experiments Setup

Implementation Details of SelfBootTok. In this section, we introduce the experiment settings of our method. Our tokenizer utilizes Dinov2 VIT [45] as the backbone and the encoder is initialized the same as pretrained Dinov2 encoder. All the experiments are conducted on the base size of Dinov2 VIT backbone with a total of 173M parameters, while the sizes of local aligners vary during the scaling experiments, including sizes of 10M, 100M, 200M, 400M and 600M. Most of our experiments are conducted with the number of latent code N = 64 and token dimension 32, while we also include comparisons of N = 32 for ablation study (See Appendix for more details). The tokenizer is trained

Table 2: **Ablation study of different local1D aligners.** We compare reconstruction performance using I-JEPA, DINOv2, and SigLIP as 1D alignment encoders, evaluated by rFID, PSNR, and SSIM metrics under the 400M local 2D aligner.

Aligner	rFID (↓)	PSNR (↑)	SSIM (↑)
I-JEPA	**0.66**	**22.86**	**0.7288**
DINOv2	0.70	22.68	0.7192
SigLIP	0.72	22.45	0.7150

Table 3: **Ablation results of different design choices** (600M parameter tokenizer, 64 tokens). G: global; L1/L2: local 1D/2D; C: causal; CB: local codebooks.

Method	C	CB	gFID	rFID
SoftVQ	–	–	1.78	0.88
G + L1	–	✓	1.70	0.76
G + L1 + L2	×	✓	1.65	0.72
G + L1 + L2	✓	×	1.68	0.73
G + L1 + L2	✓	✓	1.56	0.66

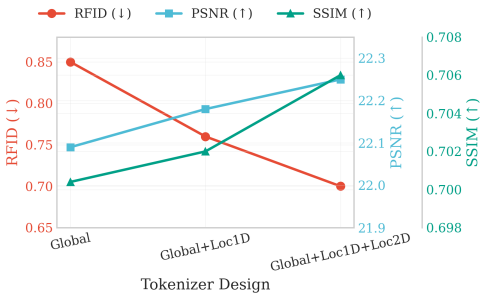

Figure 5: Ablation study of global-local self-bootstrapping design. RFID, PSNR, and SSIM curves are reported. As shown, incorporating self-bootstrapped local 1D and 2D tokens substantially improves reconstruction performance across all metrics.

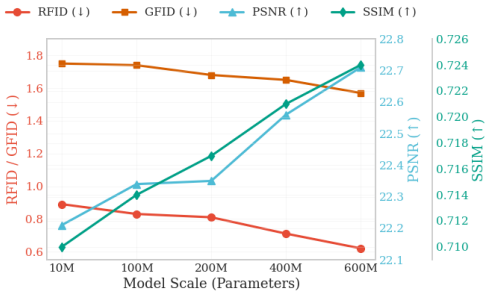

Figure 6: Scaling results for the local 2D aligner are presented, where the model size is scaled from 10M to 600M. As shown, efficient scaling improves both reconstruction and generation quality. Notably, for the five generation results (gFID) shown, the generator is trained only once, resulting in a 40 percent reduction in total computational cost.

on ImageNet at a resolution of 256×256 for 50 epochs with a small local aligner, and up to 75 epochs with larger local aligners in the scaling experiments. For discriminator training, we adopt a setup similar to SoftVQ [12], using a StyleGAN-like architecture [26, 27] and strategies such as LeCAM [53] and consistency regularization [66].

Implementation Details of Generators. We use SiT as the generator for downstream denoising-based image generation tasks. In our setup, where only global tokens need to be generated, we train our tokenizer for 50 epochs and SiT for 800K iterations. For subsequent scaling, we expand the local 2D aligner and train the tokenizer pipeline for an additional 50 to 75 epochs, keeping the ViT encoder frozen. Notably, SiT requires no further training. To accelerate SiT's training, we apply the representation disentangling method [59] and incorporate a pretrained DINOv2-base encoder [45] to enhance convergence speed.

Evaluation. We adopt reconstruction Frechet Inception Distance (rFID) [25] on ImageNet validation set to assess the performance of tokenizer. Furthermore, we include PSNR and SSIM metrics to more accurately reflect the reconstruction quality. To evaluate the generation quality, we report generation FID (gFID) under either with or without CFG setting. Additionally, we evaluate the efficiency of generative models with respect to tokenizer GFLOPs. For further details, refer to the Appendix.

4.2 Quantitative Results

We analyze the reconstruction and generation results, along with the training efficiency in this section.

High reconstruction quality with only 64 tokens. As shown in Table 1, our method achieves superior reconstruction performance (, an rFID score of **0.66**) using only 64 tokens, outperforming other baselines trained with the same token budget [12, 65] and performing comparably to methods [18, 48, 61] that rely on 256 tokens. Moreover, after efficient scaling, our method achieves better PSNR

and SSIM scores for reconstruction, as illustrated in Fig. 6. This demonstrates the potential for progressively improved reconstruction performance as the local aligner is scaled up within our architecture.

State-of-the-art generation performance among 1D tokenizers. In terms of generation, our approach achieves a state-of-the-art gFID score of **1.56** using only 64 tokens, as reported in Table 1. This result is comparable to the best-performing 2D tokenizer VA-VAE (, 1.35 gFID with 256 tokens) and the top autoregressive tokenizer MAR-H (, 1.55 gFID with 256 tokens). Moreover, our model attains a lower gFID even without classifier-free guidance (CFG), outperforming other 1D and autoregressive baselines under the same setting. This shows that incorporating multi-granularity information into our model architecture enhances generation quality, leading to superior performance and validating the effectiveness of our design.

Efficient training. Our method also exhibits strong training efficiency, achieving competitive performance without additional generation optimization overhead. As illustrated in Fig. 4, once the tokenizer pipeline is trained, our framework supports parallel optimization of the scaled local aligners and the generator. Unlike conventional approaches that require retraining the generator each time the tokenizer is scaled, our design leverages global tokens for generation, allowing the generator to be trained only once. For the overall scaling experiment, this strategy reduces training computation by about **40%** (from 4.9B parameters and 60K GFLOPs to 2.95B and 36.6K GFLOPs) and shortens training time by roughly **54%** (from 28 to 13 days). Moreover, our method scales efficiently to larger datasets, yielding greater computational savings as the generator size increases. Further details on computational analysis and data scaling are provided in the Appendix.

4.3 Qualitative Results

As shown in Fig. 2, our method generates diverse, high-quality real-world images using only 64 tokens, covering animals (e.g., dogs, tigers), food (e.g., noodles, burgers), and plants (e.g., flowers) sampled from 1000 ImageNet [16] classes. We examine token correspondence and the visual effects of local aligner scaling in Section 4.5.

4.4 Ablation Study

To justify our design choices and validate the self-bootstrapping paradigm, we conduct comprehensive ablation studies on local 1D/2D branches, 1D aligners, and fusion strategies. As shown in Table 3, our design outperforms all variants: we use an MLP for the efficient 1D local aligner (not scaled, captures coarse local info) and a transformer-based 2D local aligner (scaled, models rich spatial details). Fig. 5 illustrates that progressive integration of local structural info enhances reconstruction quality, with optimal performance when both local branches are included. Table 2 investigates local 1D token alignment with frozen visual encoders (I-JEPA, DINOv2, SigLIP): I-JEPA performs best, with DINOv2 and SigLIP comparable, indicating distinct local structural cues affect tokenizer behavior. In Table 7, we compare fusion strategies under Eq. 11: concatenation-based fusion improves reconstruction but increases decoder size/computational cost, degrading generation quality and revealing a task-dependent trade-off. Finally, Fig. 6 shows our method improves with local aligner scaling: at 600M parameters and 64 tokens, it achieves a gFID of 1.56, outperforming baselines and demonstrating architectural scalability.

4.5 Visual Analysis of Global–Local Decomposition and Scaling

Complementary global and local representations. Figure 7 compares Grad-CAM heatmaps for the global token group and the combined local 1D and 2D token groups. The local maps highlight image details that complement the global responses, including object configurations, background structure, and fine-grained silhouettes. This pattern is consistent with the intended division between global semantics and local structure. In SelfBootTok, both local branches are predicted from global tokens and contribute to decoding, allowing the generator to operate on a compact global representation. These qualitative observations complement the reconstruction ablations in Fig. 5.

Fine-grained details from local aligner scaling. Figure 8 compares reconstruction and generation results with and without local aligner scaling on ImageNet-1K at 256×256 resolution using 64 global tokens. The scaled model produces finer details in the illustrated examples, including bird



Figure 7: Grad-CAM visualization of global and local token groups. Left: input image. Middle: global token map. Right: combined local 1D and 2D token map. Local token responses highlight image details that complement the global token responses.



Figure 8: Qualitative comparison of reconstruction and generation on ImageNet-1K at 256×256 resolution using 64 global tokens. For each image pair, the left panel shows the result without local aligner scaling, and the right panel shows the result with scaling. The illustrated improvements include finer bird-eye details and human facial textures.

eyes and human facial textures. This observation complements the improvements in reconstruction and generation metrics reported in Fig. 6. Under our scaling strategy, the global representation is frozen, and the same generator is reused while the local 2D aligner is enlarged and trained. Thus, additional tokenizer capacity improves image detail without increasing the generator's token budget or retraining it.

5 Conclusion

In this paper, we propose SelfBootTok, an efficient 1D image tokenizer that decomposes image tokens into global and local groups and leverages a self-supervised bootstrapped paradigm to recover sufficient local information from global tokens. Our architecture reduces image token overlap, achieving higher reconstruction quality and better efficiency in downstream generation tasks, while its local self-bootstrapped prediction design enables strong scalability and highlights self-bootstrapping's potential for image tokenization.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[3] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.

[4] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023.

[5] 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.

[6] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22669–22679, 2023.

[7] L Lao Beyer, Tianhong Li, Xinlei Chen, Sertac Karaman, and Kaiming He. Highly compressed tokenizer can generate without training. *arXiv preprint arXiv:2506.08257*, 2025.

[8] Wenhao Chai, Enxin Song, Yilun Du, Chenlin Meng, Vashisht Madhavan, Omer Bar-Tal, Jenq-Neng Hwang, Saining Xie, and Christopher D Manning. Auroracap: Efficient, performant video detailed captioning and a new benchmark. *arXiv preprint arXiv:2410.03051*, 2024.

[9] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11315–11325, 2022.

[10] Cong Chen, Ziyuan Huang, Cheng Zou, Muzhi Zhu, Kaixiang Ji, Jiajia Liu, Jingdong Chen, Hao Chen, and Chunhua Shen. Hieratok: Multi-scale visual tokenizer improves image reconstruction and generation. *arXiv preprint arXiv:2509.23736*, 2025.

[11] Hao Chen, Yujin Han, Fangyi Chen, Xiang Li, Yidong Wang, Jindong Wang, Ze Wang, Zicheng Liu, Difan Zou, and Bhiksha Raj. Masked autoencoders are effective tokenizers for diffusion models. In *Forty-second International Conference on Machine Learning*, 2025.

[12] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28358–28370, 2025.

[13] Haozhe Chi, Zhicheng Sun, Yang Jin, Yi Ma, Jing Wang, and Yadong Mu. Enhancing consistency of flow-based image editing through kalman control. *Advances in Neural Information Processing Systems*, 38:141866–141898, 2026.

[14] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023.

[15] Quan Dao, Hao Phung, Binh Nguyen, and Anh Tran. Flow matching in latent space. *arXiv preprint arXiv:2307.08698*, 2023.

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[17] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Lowis3d: Language-driven open-world instance-level 3d scene understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):8517–8533, 2024.

[18] 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*, pages 12873–12883, 2021.

[19] Carlos Esteves, Mohammed Suhail, and Ameesh Makadia. Spectral image tokenizer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17181–17190, 2025.

[20] Yunhao Fang, Ligeng Zhu, Yao Lu, Yan Wang, Pavlo Molchanov, Jan Kautz, Jang Hyun Cho, Marco Pavone, Song Han, and Hongxu Yin. Vila^2: Vila augmented vila. *arXiv preprint arXiv:2407.17453*, 2024.

[21] Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37:133345–133385, 2024.

[22] Philippe Hansen-Estruch, David Yan, Ching-Yao Chung, Orr Zohar, Jialiang Wang, Tingbo Hou, Tao Xu, Sriram Vishwanath, Peter Vajda, and Xinlei Chen. Learnings from scaling visual tokenizers for reconstruction and generation, 2025.

[23] Dongze Hao, Qunbo Wang, Longteng Guo, Jie Jiang, and Jing Liu. Self-bootstrapped visual-language model for knowledge selection and question answering. *arXiv preprint arXiv:2404.13947*, 2024.

[24] Ju He, Qihang Yu, Qihao Liu, and Liang-Chieh Chen. Flowtok: Flowing seamlessly across text and image tokens. *arXiv preprint arXiv:2503.10772*, 2025.

[25] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[27] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[28] 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. *arXiv preprint arXiv:2501.07730*, 2025.

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

[30] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11523–11532, 2022.

[31] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024.

[32] Jiachen Li, Xinyao Wang, Sijie Zhu, Chia-Wen Kuo, Lu Xu, Fan Chen, Jitesh Jain, Humphrey Shi, and Longyin Wen. Cumo: Scaling multimodal llm with co-upcycled mixture-of-experts. *Advances in Neural Information Processing Systems*, 37:131224–131246, 2024.

[33] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[34] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

[35] Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. Mage: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2142–2152, 2023.

[36] 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.

[37] Xiang Li, Kai Qiu, Hao Chen, Jason Kuen, Jiuxiang Gu, Bhiksha Raj, and Zhe Lin. Imagefolder: Autoregressive image generation with folded tokens. *arXiv preprint arXiv:2410.01756*, 2024.

[38] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[39] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[40] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[41] Yadong Lu, Chunyuan Li, Haotian Liu, Jianwei Yang, Jianfeng Gao, and Yelong Shen. An empirical study of scaling instruct-tuned large multimodal models. *arXiv preprint arXiv:2309.09958*, 2023.

[42] Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers, 2024.

[43] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023.

[44] 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.

[45] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[47] Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Flowar: Scale-wise autoregressive image generation meets flow matching. *arXiv preprint arXiv:2412.15205*, 2024.

[48] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[49] 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*, 2024.

[50] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[51] 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.

[52] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[53] Hung-Yu Tseng, Lu Jiang, Ce Liu, Ming-Hsuan Yang, and Weilong Yang. Regularizing generative adversarial networks under limited data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7921–7931, 2021.

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

[55] Abdul Waheed, Zhen Wu, Dareen Alharthi, Seungone Kim, and Bhiksha Raj. Videojudge: Bootstrapping enables scalable supervision of mllm-as-a-judge for video understanding. *arXiv preprint arXiv:2509.21451*, 2025.

[56] Xidong Wang, Dingjie Song, Shunian Chen, Junyin Chen, Zhenyang Cai, Chen Zhang, Lichao Sun, and Benyou Wang. Longllava: Scaling multi-modal llms to 1000 images efficiently via a hybrid architecture. *arXiv preprint arXiv:2409.02889*, 2024.

[57] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *European Conference on Computer Vision*, pages 396–416. Springer, 2024.

[58] Mark Weber, Lijun Yu, Qihang Yu, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. Maskbit: Embedding-free image generation via bit tokens. *arXiv preprint arXiv:2409.16211*, 2024.

[59] Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, et al. Representation entanglement for generation: Training diffusion transformers is much easier than you think. *arXiv preprint arXiv:2507.01467*, 2025.

[60] Jiaer Xia, Bingkui Tong, Yuhang Zang, Rui Shao, and Kaiyang Zhou. Bootstrapping grounded chain-of-thought in multimodal llms for data-efficient model adaptation. *arXiv preprint arXiv:2507.02859*, 2025.

[61] Tianwei Xiong, Jun Hao Liew, Zilong Huang, Jiashi Feng, and Xihui Liu. Gigatok: Scaling visual tokenizers to 3 billion parameters for autoregressive image generation. *arXiv preprint arXiv:2504.08736*, 2025.

[62] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15703–15712, 2025.

[63] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

[64] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023.

[65] 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, 2024.

[66] Han Zhang, Zizhao Zhang, Augustus Odena, and Honglak Lee. Consistency regularization for generative adversarial networks. *arXiv preprint arXiv:1910.12027*, 2019.

[67] Yangming Zhang, Wenqi Jia, Wei Niu, and Miao Yin. GaussianSpa: An "optimizing-sparsifying" simplification framework for compact and high-quality 3D gaussian splatting. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26673–26682, 2025.

[68] Yangming Zhang, Jian Xu, Chaojian Li, Kunxiong Zhu, Wei Niu, Gagan Agrawal, Yang Katie Zhao, Jian Wang, Yingyan Celine Lin, and Miao Yin. Gaussians on a diet: High-quality memory-bounded 3D gaussian splatting training. *arXiv preprint arXiv:2604.20046*, 2026.

[69] Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023.

A Theoretical Analysis of Hierarchical Self-Bootstrapped VAE.

A.1 Preliminaries

We briefly recall the standard VAE objective and its hierarchical extension. Let $x \in \mathcal{X}$ denote an observed image, and let the latent variables be $z_G \in \mathcal{Z}_G$ (global tokens) and $z_L \in \mathcal{Z}_L$ (local tokens). A hierarchical generative model factorizes as

$$p_\theta(x, z_G, z_L) = p_\theta(x \mid z_G, z_L)\, p_\theta(z_L \mid z_G)\, p_\theta(z_G). \tag{14}$$

The variational posterior is factorized as

$$q_\phi(z_G, z_L \mid x) = q_\phi(z_G \mid x)\, q_\phi(z_L \mid x), \tag{15}$$

which may still allow cross-level dependency through shared encoders or alignment terms.

The hierarchical VAE optimizes the evidence lower bound (ELBO)

$$\begin{aligned}\mathcal{L}_B(x) =& _{q_\phi(z_G, z_L \mid x)}\,[\log p_\theta(x \mid z_G, z_L)] \\ & - (q_\phi(z_G \mid x) \| p_\theta(z_G)) \\ & - _{q_\phi(z_G \mid x)}\,(q_\phi(z_L \mid x) \| p_\theta(z_L \mid z_G)).\end{aligned} \tag{16}$$

The baseline single-latent VAE assumes a flat structure with latent variable $z \in \mathcal{Z}$,

$$\mathcal{L}_A(x) = \mathbb{E}_{q_\psi(z \mid x)}[\log p_\psi(x \mid z)] - \mathrm{KL}(q_\psi(z \mid x) \,\|\, p_\psi(z)). \tag{17}$$

A.2 Generative Assumptions

True generative process. The true data distribution $p^\star(x)$ factorizes through latent variables $z_G^\star, z_L^\star$ as

$$p^\star(x) = \int p^\star(x \mid z_G, z_L)\, p^\star(z_L \mid z_G)\, p^\star(z_G)\, dz_L\, dz_G. \tag{18}$$

Approximation quality The conditional prior in the hierarchical model $p_\theta(z_L \mid z_G)$ approximates the true conditional $p^\star(z_L \mid z_G)$ with an expected KL error

$$\delta = \mathbb{E}_{p^\star(z_G)}\big[\mathrm{KL}\big(p^\star(z_L \mid z_G) \,\|\, p_\theta(z_L \mid z_G)\big)\big]. \tag{19}$$

Decoder regularity The decoder $p_\theta(x \mid z_G, z_L)$ is L-Lipschitz in (z_G, z_L) and has bounded log-likelihood: $|\log p_\theta(x \mid z_G, z_L)| \le M$ for all (x, z_G, z_L).

A.3 Main Proposition

ELBO Advantage of Hierarchical Decomposition Under Assumptions A.2, there exists a constant $C = O(LM)$ such that

$$\sup_{\phi,\theta} \mathcal{L}_B(x) \;\ge\; \sup_{\psi} \mathcal{L}_A(x) - C\,\delta, \tag{20}$$

where $\mathcal{L}_A$ and $\mathcal{L}_B$ are defined in Eqs. (17) and (16), respectively. In other words, when the hierarchical conditional prior $p_\theta(z_L \mid z_G)$ closely matches the true dependency between global and local factors ($\delta \approx 0$), the hierarchical model achieves an equal or tighter ELBO than the monolithic VAE.

A.4 Proof Sketch

Step 1: Idealized hierarchical ELBO. Suppose we can set $p_\theta = p^\star$ and $q_\phi = p^\star(\cdot \mid x)$, then

$$\begin{aligned}\mathcal{L}_B^\star(x) &= \log p^\star(x) - _{p^\star(z_G \mid x)}\big(p^\star(z_L \mid x, z_G) \,\|\, p^\star(z_L \mid z_G)\big) \\ &\ge \log p^\star(x).\end{aligned} \tag{21}$$

since the KL term is non-negative and the ELBO is tight at equality.

Step 2: Decomposition of modeling error. For any parameterization (ϕ, θ),

$$\begin{aligned}\log p^\star(x) - \mathcal{L}_B(x) &= \big(q_\phi(z_G, z_L \mid x) \,\|\, p_\theta(z_G, z_L \mid x)\big) \\ &\le (q_G \| p_\theta(z_G)) + _{q_G}\,(q_L \| p_\theta(z_L \mid z_G)).\end{aligned} \tag{22}$$

Substituting the approximation quality of the conditional prior (Assumption A.2) gives an additive deviation bounded by δ up to a constant scaling C from the decoder's smoothness.

Step 3: Comparison with single-latent model. The single-latent VAE must represent both global and local variations in a single latent vector z. Its optimal ELBO can be expressed as

$$\sup_{q_\psi, \theta_\psi} \mathcal{L}_A(x) = \log p^\star(x) - \inf_{q_\psi, \theta_\psi} \mathrm{KL}\big(q_\psi(z \mid x) \,\|\, p_\psi(z \mid x)\big).$$

Since any $p_\psi(z)$ can be viewed as marginalizing $p_\theta(z_G, z_L)$ under a coupling $z = f(z_G, z_L)$, but with information loss (by the data-processing inequality), the marginal model p_ψ cannot achieve a smaller KL to the true posterior than the factorized one. Therefore,

$$\sup \mathcal{L}_B(x) \geq \sup \mathcal{L}_A(x) - C\delta.$$

□

A.5 Discussion and limitation

Although our method improves tokenization quality and introduces an efficient scaling paradigm for image generation, the fusion of global and local tokens demands careful architectural design, especially as token counts increase. Further work is needed to address this limitation.

A.6 Discussion of proof

The proposition above provide a theoretical justification for the proposed global–local, self-bootstrapped tokenizer. When the true generative process admits a hierarchical factorization, explicitly modeling the conditional prior yields a tighter or equal ELBO bound up to the conditional modeling error δ.

B Detailed Optimal Transport Formulation for 2D to 1D Alignment

This section provides a complete theoretical formulation of the 2D–to–1D alignment process using entropy-regularized optimal transport (OT), the existence and uniqueness of the Sinkhorn solution, and the differentiability properties required for end-to-end training.

B.1 Problem Setup

Let $\mathbf{X} \in \mathbb{R}^{N_x \times d}$ denote the 2D VAE latent tokens ($N_x = H \times W$), and let $\mathbf{Z} \in \mathbb{R}^{N_z \times d}$ denote the predicted 1D latent tokens. We consider probability vectors

$$\mathbf{a} \in \Delta^{N_x}, \qquad \mathbf{b} \in \Delta^{N_z},$$

where $\Delta^n = \{\mathbf{u} \in \mathbb{R}^n_+ : \sum_i u_i = 1\}$.

We define the quadratic ground cost

$$C_{ij} = \|\mathbf{x}_i - \mathbf{z}_j\|_2^2, \qquad 1 \leq i \leq N_x,\ 1 \leq j \leq N_z. \tag{23}$$

Let $\Pi(\mathbf{a}, \mathbf{b})$ denote the transport polytope:

$$\Pi(\mathbf{a}, \mathbf{b}) := \big\{\mathbf{P} \in \mathbb{R}_+^{N_x \times N_z} :\ \mathbf{P}\mathbf{1} = \mathbf{a},\ \mathbf{P}^\top \mathbf{1} = \mathbf{b}\big\}.$$

B.2 Entropy-Regularized Optimal Transport

The alignment objective is formulated as the entropy-regularized OT problem

$$\mathbf{P}^\star = \arg\min_{\mathbf{P} \in \Pi(\mathbf{a}, \mathbf{b})} \Big[\langle \mathbf{P}, \mathbf{C} \rangle + \varepsilon\, \mathcal{H}(\mathbf{P})\Big], \tag{24}$$

where the entropy term is

$$\mathcal{H}(\mathbf{P}) = \sum_{i,j} P_{ij}(\log P_{ij} - 1).$$

The regularization parameter $\varepsilon > 0$ ensures strict convexity, which guarantees a unique minimizer.

Fenchel Dual Form. The dual of (24) is

$$\max_{\alpha,\beta\in\mathbb{R}} \alpha^\top \mathbf{a} + \beta^\top \mathbf{b} - \varepsilon \sum_{i,j} \exp\left(\frac{\alpha_i + \beta_j - C_{ij}}{\varepsilon}\right).$$

Strong duality holds because the primal is strictly convex over a compact polytope.

B.3 Sinkhorn Solution

Define the Gibbs kernel

$$K_{ij} = \exp(-C_{ij}/\varepsilon). \tag{25}$$

The Sinkhorn algorithm alternates row and column normalizations:

$$\mathbf{P}^{(t)} = \mathrm{diag}(\mathbf{u}^{(t)})\, \mathbf{K}\, \mathrm{diag}(\mathbf{v}^{(t)}),$$

with updates

$$\mathbf{u}^{(t+1)} = \frac{\mathbf{a}}{\mathbf{K}\mathbf{v}^{(t)}}, \quad \mathbf{v}^{(t+1)} = \frac{\mathbf{b}}{\mathbf{K}^\top \mathbf{u}^{(t+1)}}.$$

Theorem 1 (Existence and Uniqueness). *For any strictly positive kernel* $\mathbf{K}$ *and marginals* $\mathbf{a}, \mathbf{b} \in \Delta$*, the Sinkhorn iterations converge to unique scaling vectors* $(\mathbf{u}^\star, \mathbf{v}^\star)$ *such that*

$$\mathbf{P}^\star = \mathrm{diag}(\mathbf{u}^\star)\mathbf{K}\mathrm{diag}(\mathbf{v}^\star)$$

is the unique minimizer of the entropy-regularized OT problem (24)*.*

Proof Sketch. The kernel $\mathbf{K}$ is strictly positive by (25). Entropy regularization makes the objective strictly convex on the transport polytope. By the Sinkhorn–Knopp theorem, the iterative proportional fitting procedure converges to the unique matrix in the matrix scaling set matching $\mathbf{a}$ and $\mathbf{b}$. □

B.4 Transport Cost as Alignment Loss

The OT alignment loss used in the main paper is

$$\mathcal{L}_{\mathrm{OT}} = \langle \mathbf{P}^\star, \mathbf{C} \rangle. \tag{26}$$

This promotes geometric agreement between the 2D latent manifold and the 1D latent sequence.

B.5 Differentiability and Gradient Computation

Entropy-regularized OT is smooth in the cost matrix C (Cuturi 2013). Let $f(C) = \langle P^\star(C), C \rangle$. The gradient is

$$\frac{\partial f}{\partial C} \;=\; \mathbf{P}^\star.$$

Thus, gradients flow from the OT plan into both $\mathbf{X}$ and $\mathbf{Z}$:

$$\frac{\partial \mathcal{L}_{\mathrm{OT}}}{\partial \mathbf{x}_i} = 2 \sum_j P^\star_{ij}(\mathbf{x}_i - \mathbf{z}_j), \qquad \frac{\partial \mathcal{L}_{\mathrm{OT}}}{\partial \mathbf{z}_j} = 2 \sum_i P^\star_{ij}(\mathbf{z}_j - \mathbf{x}_i).$$

These closed-form expressions ensure efficient backpropagation.

B.6 Stability and Regularization Properties

Entropy Regularization. The entropic term guarantees:

- strictly convex objective,
- unique optimal transport plan,
- improved numerical stability,
- differentiability of the mapping $(C, a, b) \mapsto P^\star$.

Sharpness vs. Smoothness. As $\varepsilon \to 0$, the plan converges to the classical OT solution. Larger ε spreads mass more broadly and stabilizes gradients.

B.7 Summary

The 2D to 1D alignment objective is fully characterized by the entropy-regularized optimal transport problem (24), solved uniquely by the Sinkhorn iterations, and the alignment loss (26) provides a stable and fully differentiable measure of geometric consistency between 2D VAE latents and predicted 1D token sequences.

C Additional Details of Implementation.

This section provides a comprehensive description of the implementation settings used for training the proposed tokenizer.

General Configuration. We fix the global random seed to 42 to ensure reproducibility. Training logs are recorded every 50 iterations, visualizations every 5000 iterations, and checkpoints every 3000 iterations. Mixed-precision training is enabled using bf16. Compiler-level optimizations are disabled (compile = False).

Dataset and Preprocessing. All experiments are conducted on the ImageNet. Input images are resized to a spatial resolution of 256×256.

Tokenizer and Vector Quantization. The temperature parameter is set to $\tau = 0.07$, and EMA updates for codebook vectors are enabled. The tokenizer employs several codebooks each containing 8192 entries, each with embedding dimension 32. The following loss components are used during training:

- entropy regularization weight: 0.01,
- VQ quantization loss weight: 0.0,
- commitment loss coefficient: 0.0,
- reconstruction loss weight: 1.0,
- perceptual loss weight: 1.0 with a warm-up of the first 5000 iterations.

Adversarial training is performed with a discriminator weight of 0.2. Diffusion-based augmentation (DiffAug) is enabled. The discriminator incorporates a consistency-regularization term weighted by 4.0 and is activated after 20000 iterations. We employ a DINO-based discriminator using hinge losses for both generator and discriminator; the LeCAM stabilization term is applied with weight 0.001.

Encoder and Decoder Architecture. Both encoder and decoder are implemented using Vision Transformers (ViTs):

- **Encoder:** We use the vit_base_patch14_dinov2.lvd142m model with a patch size of 16, initialized with pretrained weights, and fully fine-tuned during training.
- **Decoder:** We employ the same ViT architecture but without pretrained initialization. The patch size is also 16, and full tuning is applied. The decoder outputs 128 latent tokens.

REPA Representation Alignment. Representation alignment is enabled via the REPA module. The REPA backbone is a vit_large_patch14_dinov2.lvd142m transformer with patch size 16 for global repa and local 1D repa, while the backbone turns to SIGLIP2 and I-JEPA for local 2D repa. The projected feature dimension is set to 1024. The REPA loss is weighted by 0.1, and alignment is performed using the repeat strategy.

Optimization Settings. Training is conducted for 50 epochs using the AdamW optimizer with a learning rate of 1×10^{-4}. The learning rate follows a cosine decay schedule. The weight decay is 1e-4, and the AdamW momentum parameters are set to $\beta_1 = 0.9$ and $\beta_2 = 0.95$. We apply gradient clipping with a maximum global norm of 1.0. The global batch size is set to 700.

Table 4: Data scaling results on RFID, GFID, PSNR, and SSIM.

Setting	RFID	GFID	PSNR	SSIM
Before Data Scaling	0.66	1.56	22.86	0.7288
After Data Scaling	0.63	1.53	22.88	0.7289

Table 5: Comparison of 32-token and 64-token settings using our method.

Setting	RFID (↓)	PSNR (↑)	SSIM (↑)
32-token	0.87	20.96	0.6993
64-token	0.66	22.86	0.7288

D Additional Results for Computational Analysis.

In this section, we provide further details on the computational analysis of our proposed method. The base model without the local 2D aligner contains **250M** parameters, while the local 2D aligner is scaled across **100M**, **200M**, **400M**, and **600M** configurations. The SiT generator includes **650M** parameters. Therefore, the total number of parameters involved in the scaling experiment is computed as:

$$\begin{aligned} &250\text{M} \times 4 + (100\text{M} + 200\text{M} + 400\text{M} + 600\text{M}) \\ &\quad + \ 650\text{M} \times 4 \\ &= 4{,}900\text{M}. \end{aligned}$$

In contrast, our pipeline trains the generator only once, which reduces the total parameter count to:

$$\begin{aligned} &250\text{M} \times 4 + (100\text{M} + 200\text{M} \\ &\quad + 400\text{M} + 600\text{M}) + 650\text{M} \\ &= 2{,}950\text{M}. \end{aligned}$$

This modification also decreases the total training GFLOPs from **60K** to **36.6K**, corresponding to an approximate **40%** reduction in computational cost.

Furthermore, training each tokenizer scale requires approximately **2 days**, and training the generator requires **5 days**. Consequently, the total training time is reduced from

$$(2 \times 4) + (5 \times 4) = 28 \text{ days}$$

to

$$(2 \times 4) + 5 = 13 \text{ days},$$

yielding an approximate **54%** reduction in overall training time.

E Additional Quantitative Results and Data Scaling Analysis

In this section, we provide additional quantitative results together with a detailed analysis of data scaling—that is, assessing the behavior of our method when trained on a substantially larger dataset. As shown in Table 4, incorporating an additional dataset of one million images leads to consistent improvements in both reconstruction and generation quality. Moreover, our method is compatible with different token capacities; , the 32-token and 64-token configurations in Table 5 exhibit stable performance, demonstrating the flexibility of our tokenizer design. In addition, the scalability of our approach is further validated in Table 9, where all reconstruction and generation metrics improve monotonically as the model size increases. Finally, the impact of our local aligner modules is highlighted in Table 8. The introduction of local 1D and 2D aligners leads to substantial gains in reconstruction fidelity, underscoring the importance and effectiveness of the proposed architectural design.

F Additional Qualitative Results.

In this section, we present additional qualitative results using a broader selection of ImageNet classes. We provide examples from twelve additional categories and illustrate the corresponding generated

Table 6: Efficiency comparison of computational overhead at generation time.

Method	Generator (650M parameters)	gFID	GFLOPs	Latency
SoftVQ	SiT(650M)	1.78	7.3K	7.5ms
Ours	generative component(250M)+SiT(400M)	1.56	7.1K	5.2ms

Table 7: **Ablation study of different fusion strategies.** Comparison between concatenation-based and soft residual fusion under the 400M local 2D aligner and I-JEPA local 1D aligner settings.

Fusion Strategy	rFID (↓)	PSNR (↑)	SSIM (↑)
Concatenation Fusion	**0.66**	**22.86**	**0.7288**
Soft Residual Fusion	0.71	22.56	0.7210

images. As shown, our method produces high-fidelity and semantically coherent natural images across diverse classes, further demonstrating its strong generative capability.



Figure 9: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "bath towel" (class ID 434).

G Appendix References

[1] Kingma, D. P., & Welling, M. Auto-encoding variational Bayes. *ICLR*, 2014.

[2] Rezende, D. J., Mohamed, S., & Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *ICML*, 2014.

[3] Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., & Winther, O. Ladder variational autoencoders. *NeurIPS*, 2016.

[4] Wainwright, M. J. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint.* Cambridge University Press, 2019.

[5] Boucheron, S., Lugosi, G., & Massart, P. *Concentration Inequalities: A Nonasymptotic Theory of Independence.* Oxford University Press, 2013.

Table 8: Comparison of tokenizer designs. Lower RFID and higher PSNR/SSIM are better.

Design	RFID (↓)	PSNR (↑)	SSIM (↑)
Global	0.85	22.09	0.7004
Global + Loc1D	0.76	22.18	0.7020
Global + Loc1D + Loc2D	0.70	22.25	0.7060

Table 9: Performance of the tokenizer at different model scales. Lower RFID/GFID and higher PSNR/SSIM indicate better performance.

Model Scale	RFID (↓)	GFID (↓)	PSNR (↑)	SSIM (↑)
10M	0.89	1.75	22.21	0.7103
100M	0.83	1.74	22.34	0.7141
200M	0.81	1.68	22.35	0.7172
400M	0.71	1.65	22.56	0.7211
600M	0.66	1.56	22.86	0.7288



Figure 10: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "balloon" (class ID 417).



Figure 11: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "tree-frog" (class ID 31).

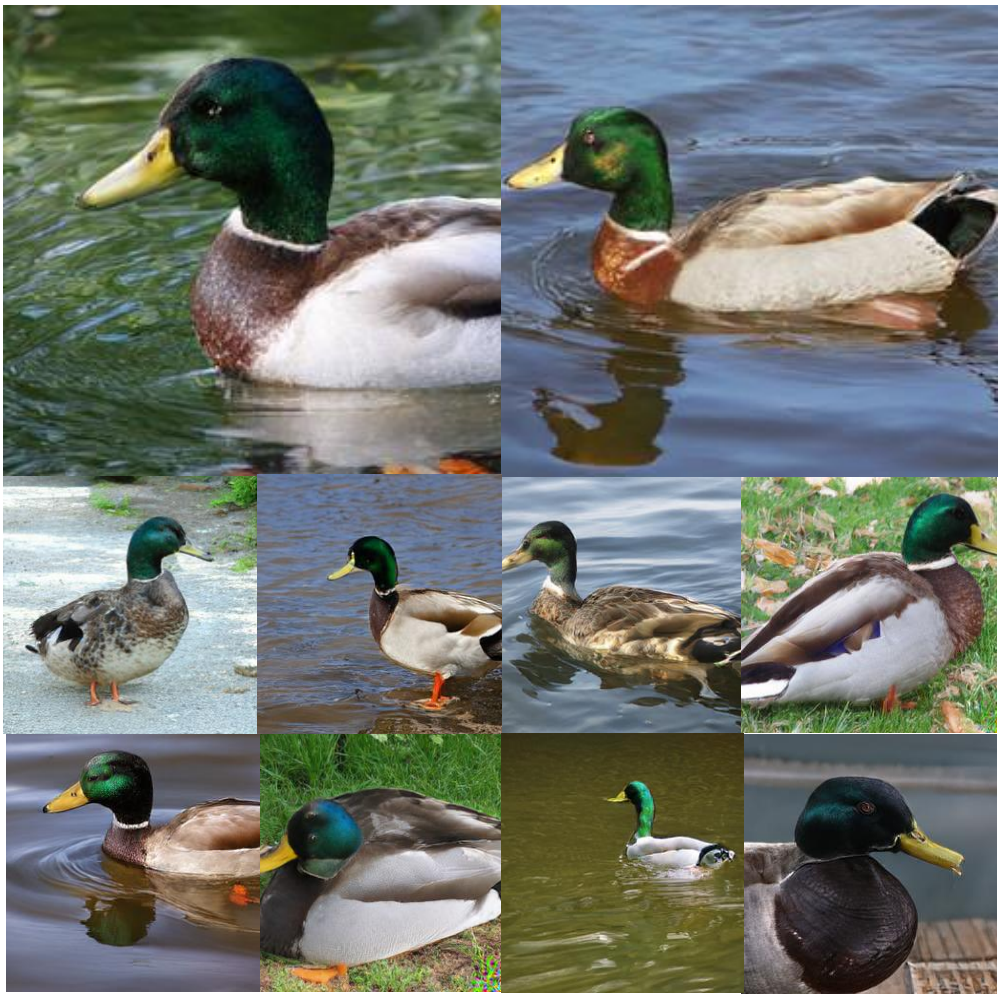

Figure 12: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is “drake” (class ID 97).



Figure 13: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is “macaw” (class ID 88).



Figure 14: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "tarantula" (class ID 76).



Figure 15: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is “mud turtle” (class ID 35).

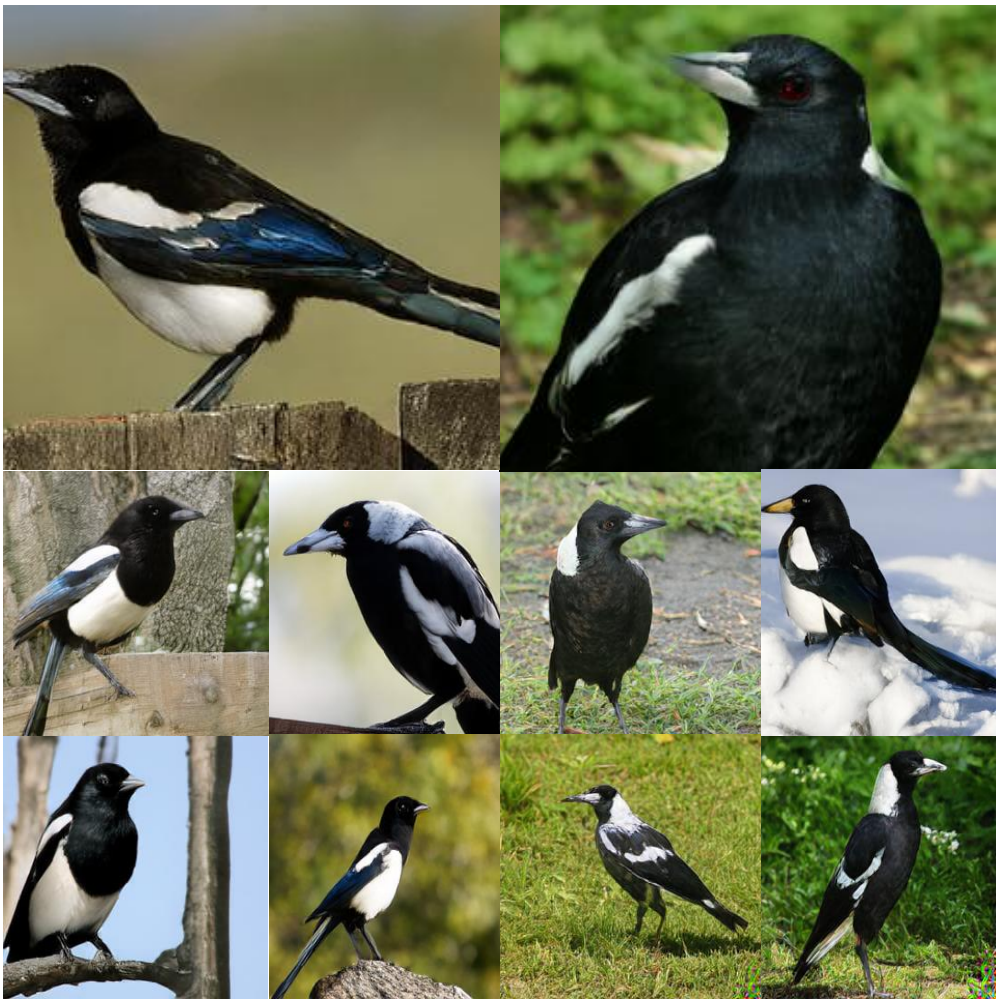

Figure 16: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "magpie" (class ID 18).



Figure 17: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "jacamar" (class ID 95).



Figure 18: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is “silky terrier, Sydney silky” (class ID 201).



Figure 19: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is “barn” (class ID 425).



Figure 20: 256×256 generation results produced by SiT-XL/1 using SelfBootTok with 64 tokens. We apply classifier-free guidance (CFG) with a scale of 2.3. The class label used for sampling is "alligator lizard" (class ID 44).

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, claims reflect our contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, we discuss the limitation.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Yes, we provide detailed proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes, we have disclosed all the information.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [N/A]

   Justification: The code is being collected and profiling.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, we include it in experiment section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our work does not require Experiment statistical significance study.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes, we discuss it in experiment section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, we follow the code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Yes, we discuss the broader impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Yes, we have described it.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes, all creators are properly credited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing experiments or human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No study of participants.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: No LLM usage in the core contribution.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.