End-to-End Vision Tokenizer Tuning

Wenxuan Wang[1,2,3*], **Fan Zhang**[3*], **Yufeng Cui**[3*], **Haiwen Diao**[4,3*],
Zhuoyan Luo[5,3], **Huchuan Lu**[4], **Jing Liu**[1,2†], **Xinlong Wang**[3†]
[1]Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Beijing Academy of Artificial Intelligence
[4]Dalian University of Technology [5] Tsinghua University
{wangwenxuan2023@ia.ac.cn,zhangfan@baai.ac.cn,wangxinlong@baai.ac.cn}

Abstract

Existing vision tokenization isolates the optimization of vision tokenizers from downstream training, implicitly assuming the visual tokens can generalize across various tasks, *e.g.*, image generation and visual question answering. The vision tokenizer optimized for low-level reconstruction is agnostic to downstream tasks requiring varied representations and semantics. This decoupled paradigm introduces a critical misalignment: The loss of the vision tokenization can be the representation bottleneck for target tasks. For example, errors in tokenizing text in an image lead to poor results when recognizing or generating them. To address this, we propose **ETT**, an end-to-end vision tokenizer tuning approach that enables joint optimization between vision tokenization and target autoregressive tasks. Unlike prior autoregressive models that use only discrete indices from a frozen vision tokenizer, **ETT** leverages the visual embeddings of the tokenizer codebook, and optimizes the vision tokenizers end-to-end with both reconstruction and caption objectives. Our **ETT** is simple to implement and integrate, without the need to adjust the original codebooks or architectures of large language models. Extensive experiments demonstrate that our end-to-end vision tokenizer tuning unlocks significant performance gains, *i.e.*, 2-6% for multimodal understanding and visual generation tasks compared to frozen tokenizer baselines, while preserving the original reconstruction capability. We hope this very simple and strong method can empower multimodal foundation models besides image generation and understanding.

1 Introduction

Recently, the rapid advancement of large language models (LLMs) and multimodal pre-training has propelled autoregressive (AR) modeling beyond its dominance in natural language processing, extending its influence into the vision and multimodal tasks. Under the next-token prediction (NTP) paradigm of autoregressive models, multimodal learning typically encodes multimodal data such as images and text into compact discrete tokens for unified sequence modeling. For example, a recent work Emu3 [54] tokenizes text, images, and video into discrete tokens and performs next-token prediction in a unified token space. Numerous subsequent works [37, 61, 69, 57], have further advanced this direction, achieving improved performance in visual generation and perception. This unified next-token prediction paradigm enables a flexible multimodal learning framework for both training and inference at scale.

Tokenization plays a key role in an autoregressive framework. For modalities such as image and video, training an efficient and general-purpose tokenizer raises significant challenges as the tokenization can

*Equal contribution. † Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

https://doi.org/10.52202/085713-1589

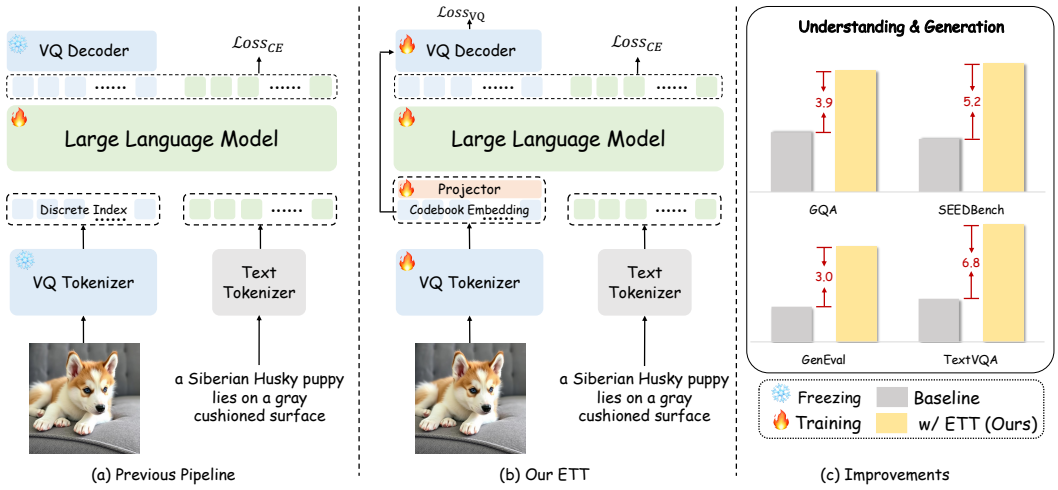

Figure 1: **Left:** Existing autoregressive pipeline uses the discrete indices from a frozen vision tokenizer optimized with low-level reconstruction. **Middle:** We present **ETT**, an end-to-end tokenizer tuning approach which takes advantage of the visual codebook embeddings and optimizes the vision tokenizer and downstream training jointly. **Right:** Our proposed **ETT** unlocks significant performance gains on multimodal understanding and generation benchmarks.

hardly be compact and lossless at the same time. The loss of the tokenization can be the bottleneck for target tasks. For example, errors in tokenizing text in an image lead to poor results when recognizing or generating them. However, existing tokenization approaches neglect this misalignment, and train a vision tokenizer separately and directly integrate it in the downstream training, assuming that the obtained visual tokens can generalize well across tasks. The vision tokenizer optimized for autoencoding is agnostic to the downstream tasks that require various representations and semantics. For instance, most tokenizers focus on low-level pixel-wise reconstruction. The quality of the learned representations is inherently limited by the information loss caused by vector quantization (VQ), leading to inferior performance in visual understanding tasks compared to models using continuous high-level representations like CLIP [38]. Besides, existing autoregressive pipelines typically use only the discrete indices from the vision tokenizer, associated with random initialization of visual embeddings in large language models for realizing downstream tasks. This poses challenges to learn vision representations and vision-language alignment, which are crucial for multimodal learning.

In this work, we present an end-to-end vision tokenizer tuning approach (**ETT**), which enables joint optimization of vision tokenization and target autoregressive downstream tasks. Inspired by recent vision-language models [47, 46, 27, 26] that update their continuous vision encoders during training to optimize the visual representations and vision-language alignment for better visual understanding, we propose to tune the discrete vision tokenizer in an end-to-end manner and leverage large language models as the vision tokenizer's visual assistant. Specifically, we introduce the vision tokenizer's codebook embeddings instead of solely using discrete indices, and integrate token-level caption loss to optimize the representations of the vision tokenizer. Our results highlight that **ETT** significantly boosts the downstream performance of multimodal understanding and visual generation tasks, demonstrating improved discriminative and generative representations in the vision tokenizer. To be noticed, **ETT** can preserve the vision tokenizer's original image reconstruction performance. Our **ETT** is also simple to implement and integrate, without the need to adjust the LLM's original text codebook or expand its embedding layers and classification heads to learn visual embeddings and vision-language alignment from scratch.

Our main contributions can be summarized as follows:

- We present a new vision tokenizer training paradigm to unlock the vision tokenizer for downstream autoregressive tasks. The vision tokenizer is aware of and is optimized for downstream training.
- We introduce a simple yet effective approach **ETT** for end-to-end vision tokenizer tuning. **ETT** leverages the tokenizer's codebook embeddings instead of only using discrete indices, and apply token-level caption loss to optimize the representation of the vision tokenizer.

https://doi.org/10.52202/085713-1589

- **ETT** greatly improves the downstream results in next-token prediction paradigm, for both multi-modal understanding and generation, while maintaining the tokenizer's reconstruction performance.

2 Related Work

Vision Tokenizer. A vision tokenizer quantizes an image or video into discrete tokens while preserving high reconstruction quality. Specifically, VQ-VAE [53] incorporates a quantizer within an auto-encoder framework, where the quantizer learns to map continuous features into discrete representations. VQGAN [10] improves the reconstruction quality via integrating perceptual loss and adversarial loss. MoVQ [70] proposes to incorporate the spatially conditional normalization to modulate the quantized vectors, generating images with high fidelity. Some other studies focus on improving codebook utilization ratio [71, 63], or incorporating more advanced quantization techniques [68, 35, 65, 25]. Recent works [61, 69, 37, 58] make efforts to integrate semantic information into the tokenizer to improve visual representations. Quantized indices are typically used in downstream tasks, while the tokenizer itself remains frozen during downstream training. In our work, we adopt IBQ [43] as the strong baseline and leverage the codebook embeddings instead of the conventional discrete indices for end-to-end tuning.

Tokenization for Visual Generation & Understanding. Discrete visual generation has made great progress in recent years. Some works [40, 64, 45, 62, 21] employ autoregressive approaches to generate images or videos by predicting tokens sequentially. While some others, such as MaskGIT [5] and Muse [4], adopt masked vision token modeling for image generation. Recently, a few studies [54, 48, 59, 58, 55] have explored unifying visual understanding and generation within a single model with the frozen discrete vision tokenizers. For example, Emu3 [54] unifies video, image, and text in token space and achieves strong understanding and generation abilities across modalities via next-token prediction. Show-o [59] proposes to incorporate mask image modeling for image generation within an autoregressive model. Janus [55] introduces two visual encoders for understanding and generation tasks separately to alleviate intrinsic conflicts between two tasks. In this work, we focus on the next-token prediction paradigm and propose to optimize both the vision tokenizer and the autoregressive models jointly for improved visual representation, promoting downstream multimodal generation and perception tasks.

3 Methodology

3.1 Vision Tokenizer

Preliminary. A VQ-based vision tokenizer includes an encoder, a quantizer and a decoder. The encoder projects an input image $I \in \mathbb{R}^{H\times W\times 3}$ to a feature map $f \in \mathbb{R}^{h\times w\times D}$, in which D is the feature dimension and $h = H/s, w = W/s$ with s as the downsampling factor. The quantizer maps each feature in f to the nearest code from codebook $B \in \mathbb{R}^{K\times D}$ to get quantized embeddings $z \in \mathbb{R}^{h\times w\times D}$, where K is the codebook size. Finally, the decoder reconstructs the quantized embeddings back into the original image. The traditional VQ models often struggle in scaling both codebook size and code dimension. As one of the pioneering works, IBQ [43] presents the first attempt to expand the code dimension to 256 while maintaining an extreme large codebook size (*i.e.*, 262,144) by updating the entire codebook simultaneously at each training step.

Vision Tokenizer in ETT. We primarily adopt the framework of IBQ [43] for image tokenization, using a downsampling factor of $s = 16$. Each discrete token in codebook has the dimension of $D = 256$. Building upon the original IBQ, we adjust the codebook size to 131,072. The loss function $\mathcal{L}_{vq}$ for tokenizer training is:

$$\mathcal{L}_{vq} = \mathcal{L}_{rec} + \mathcal{L}_{quant} + \mathcal{L}_{lpips} + \lambda_G \cdot \mathcal{L}_{GAN} + \lambda_E \cdot \mathcal{L}_{entropy} \tag{1}$$

where $\mathcal{L}_{rec}$ is the pixel reconstruction loss, $\mathcal{L}_{quant}$ is the quantization loss between quantized embeddings and encoded features, $\mathcal{L}_{lpips}$ is the perceptual loss from LPIPS [67], $\mathcal{L}_{GAN}$ is the adversarial loss from a PatchGAN [18] and $\mathcal{L}_{entropy}$ is the entropy loss [65]. λ_G and λ_E are the weight of adversarial loss and entropy loss separately.

https://doi.org/10.52202/085713-1589

3.2 End-to-End Vision Tokenizer Tuning

Discrete Indices to Codebook Embeddings. Methods like Emu3 [54] and similar approaches such as [48], which use only the discrete indices of the vision tokenizer in downstream tasks, discard the rich representational capacity of vision tokenizer embeddings. By depending only on discrete codebook indices, these methods prevent gradient propagation, making end-to-end training infeasible. To address this limitation, we present **ETT**, which directly connects codebook embeddings from vision tokenizer to the LLM, effectively leveraging the richer feature representations encoded within the vision tokenizer while enabling end-to-end training.

LLM Bridges End-to-End Tuning. Specifically, as illustrated in Figure 1, given an input image $I \in \mathbb{R}^{H\times W\times 3}$, we first obtain its quantized embeddings $z \in \mathbb{R}^{h\times w\times D}$ from the tokenizer's codebook. To ensure compatibility with the pretrained LLM, we employ a multilayer perceptron with the GeLU activation as a lightweight projector. This projector layer maps the quantized visual embeddings z to $x^I \in \mathbb{R}^{h\times w\times C}$, where C denotes the hidden dimension size of the large language model. Since the entire computation graph, including both the pretrained LLM and the vision tokenizer, remains differentiable, the whole structure can be trained end-to-end using gradient-based optimization. For text input T, we utilize the tokenizer and text embedding layer from the pretrained LLM to convert it into text token embeddings $x^T \in \mathbb{R}^{N\times C}$.

Preservation of Reconstructive Capability. While end-to-end training enhances the representations of the vision tokenizer, it is essential to maintain its reconstructive capability to ensure high-fidelity image synthesis. To achieve this, we set the overall training objective as the combination of caption loss $\mathcal{L}_{cap}$ and VQ loss $\mathcal{L}_{vq}$. Specifically, we feed both the image token embeddings x^I and text token embeddings x^T into the LLM. For text tokens, we apply the cross-entropy (CE) loss:

$$\mathcal{L}_{cap} = -\sum_{t=1}^{N} \log P(x_t^T | x^I, x_{<t}^T) \tag{2}$$

Besides, we directly reuse the loss function L_{vq} for visual reconstruction. Thus, our end-to-end vision tokenizer tuning objective becomes:

$$\mathcal{L} = \mathcal{L}_{cap} + \alpha \cdot \mathcal{L}_{vq} \tag{3}$$

where α is the loss weight that controls the trade-offs between multimodal perception and visual reconstruction. By jointly training the tokenizer encoder and decoder with LLM, our approach maintains the model's reconstructive capability while ensuring that learned visual tokens remain semantically meaningful and effective for multimodal understanding and generation.

3.3 Training Recipe for Multimodal Generation and Understanding

Following previous works [7, 8], the whole training process for downstream multimodal perception and generation follows three sequential training stages. The employed training data comprises publicly available image datasets supplemented with diverse instruction data for understanding and generation, as shown in Table 1.

Table 1: **Details of training data across all stages for multimodal generation and perception.**

Stage	Dataset	#Num	Total
Stage 1: Alignment Learning	SOL-recap	12.0M	12.0M
Stage 2: Semantic Learning	SOL-recap	12.0M	12.0M
Stage 3: Post-Training Chat	SOL-recap	32.0M	67.3M
	LLaVA-OneVision [27]	3.5M	
	Infinity-MM [15]	31.8M	
Stage 3: Post-Training Gen	AI-generated Data	14.0M	30.0M
	High-Aesthetics Web Data	16.0M	

Stage 1: Alignment learning. The first training stage is to effectively establish the vision-language alignment. With the pretrained large language model and vision tokenizer, we keep them frozen and train only the visual projector layer with image-to-text caption loss $\mathcal{L}_{cap}$. This setup enables the

LLM to acquire visual concepts and entities directly from the tokenizer, effectively bridging vision and language modalities. Specifically, we curate a 12M-image subset from our constructed dataset SOL-recap, which comprises 32M image-text pairs sourced from publicly available datasets, *i.e.*, SA-1B [20], OpenImages [22], and LAION [42]. To be noticed, all images are recaptioned using an improved captioning engine following [8]. The high-quality data at this stage can enhance training stability and cross-modality alignment.

Stage 2: Semantic Learning. The second stage acts as the most critical part in the entire training pipeline, realizing end-to-end vision tokenizer tuning. At this stage, we unfreeze the weights of the LLM, projector, and vision tokenizer, optimizing them using the caption loss $\mathcal{L}_{cap}$ and the reconstruction loss $\mathcal{L}_{vq}$ jointly, as defined in Equation 3. The high-quality subset, *i.e.*, 12M image-text pairs from SOL-recap, is utilized for multimodal understanding and reconstruction learning. This stage enables efficient learning of the vision tokenizer's perceptual capability, supporting both visual reconstruction and understanding. This well designed stage 2 can enhance alignment between vision tokenizer and downstream tasks while preserving the original reconstructive ability.

Stage 3: Post-Training. After acquiring the enhanced vision tokenizer via the proposed end-to-end vision tokenizer tuning, we follow the standard post-training pipeline to realize multimodal understanding and generation. During this training stage, we further post-train two specialist models, *i.e.***ETT-Chat** and **ETT-Gen**, by freezing the vision tokenizer part, and tuning the visual projector, as well as the large language model layers to enhance instruction-following capabilities for multimodal understanding and text-to-image generation, respectively. For multimodal understanding, we collect a diverse set of high-quality, multi-source instruction data, including SOL-recap, LLaVA-OneVision [27], and Infinity-MM [15]. For visual generation, we construct 14M AI-generated samples using the Flux model [23] and additionally curate 16M image-text pairs from open-source web data [12, 3], filtering them based on image resolution and LAION aesthetic score [24].

4 Experimental Results

4.1 Training Settings

Data Preparation. *(1) Vision-Language Pre-training & Vision Tokenizer Datasets.* We adopt the pre-processing pipeline [8] to refine SA-1B [20], OpenImages [22], and LAION [42], resulting in 11M, 7M, and 14M images respectively. We utilize the caption engine following [8] to produce 32M high-quality captions. *(2) Supervised Fine-tuning Datasets.* For understanding datasets, we extract 31.8M multi-task samples from Infinity-MM [15] and 3.5M instruction data from LLaVA-OneVision [27], prioritizing complex conversational structures. For generation datasets, we generate 14M AI-created samples with the Flux model [23] and further select 16M image-text pairs from open-source web data [12, 3], applying filters based on image resolution and aesthetic scores [24].

Implementation Details. We train **ETT** on 8-A100 nodes using the Adam optimizer [19]. The batch sizes for Stages 1, 2, and 3 are set to 1024, 1024, and 1024, respectively, with maximum learning rates of 4×10^{-5}, 4×10^{-5}, and 2×10^{-5}. We apply a warm-up strategy with a 0.03 ratio and use a cosine decay scheduler across all stages. Unless otherwise specified, images are processed at a resolution of 512^2, and ablation studies are reported using LLaVA-mix-665K [31] at Stages 3. For all the experiments in our work, we adopt Qwen2.5-1.5B [50] as the large language model for multimodal sequence modeling.

For the vision tokenizer in **ETT**, we employ Adam optimizer [19] with a fixed learning rate of 1×10^{-4}, $\beta_1 = 0.5$ and $\beta_2 = 0.9$. The tokenizer is trained for 500,000 steps with a global batch size of 256 and an input resolution of 256×256. The adversarial loss weight λ_G is set to 0.1, and the entropy loss weight λ_E is set to 0.05. We also adopt LeCAM regularization [52] for discriminator training to improve training stability.

4.2 Multimodal Understanding Evaluation

We validate **ETT** on various widely known vision-language perception benchmarks, covering task-specific evaluations (GQA [17] and TextVQA [44]), hallucination detection (POPE [29]), open-domain multimodal understanding (MME [11], MMBench [33], SEED-Bench [28], and MM-Vet [66]), and scientific reasoning (ScienceQA-IMG [34]).

https://doi.org/10.52202/085713-1589

Table 2: **Comparison with existing state-of-the-art vision-language models on various multi-modal understanding benchmarks**, including MMB^{EN}: MMBench-EN [33]; $SEED^{I}$: SEEDBench-Img [28]; MMV: MMVet [66]; MME [11]; POPE [29]; GQA [17]; SQA^{I}: ScienceQA-Img [34]; TQA: TextVQA [44]. Note that #LLM-Param denotes the number of LLM parameters, #Data represents the pre-training / fine-tuning data volume, * denotes that the images of related training datasets are observed during training, and the best results are marked in **bold**.

Method	#LLM-Param	#Data	MMB^{en}	$SEED^{I}$	MMV	MME	POPE	GQA	SQA^{I}	TQA
Continuous VLMs:										
QwenVL-Chat [1]	7B	7.2B / 50M	60.6	58.2	-	1848.0	-	57.5	68.2	61.5
EVE [7]	7B	33M / 1.8M	52.3	64.6	25.7	1628.0	85.0	62.6	64.9	56.8
Cambrian [51]	7B	10B+ / 7M	**75.9**	**74.7**	-	-	-	**64.6**	80.4	**71.7**
LLaVA-1.5 [31]	7B	0.4B+ / 665K	64.3	64.3	30.5	**1859.0**	85.9	62.0	66.8	46.1
LLaVA-1.6 [32]	7B	0.4B+ / 760K	67.4	64.7	**43.9**	1842.0	86.4	64.2	70.2	64.9
Janus [55]	1.3B	- / -	69.4	63.7	34.3	1338.0	**87.0**	59.1	-	-
Discrete VLMs:										
Chameleon [48]	7B	1.4B+ / 1.8M	31.1	30.6	8.3	170.0	-	-	47.2	4.8
LWM [30]	7B	1B+/-	-	-	9.6	-	75.2	44.8	-	-
VILA-U [58]	7B	700M/7M	-	-	33.5	-	85.8	60.8	-	60.8
Emu3 [54]	8B	- / -	58.5	68.2	37.2	-	85.2	60.3	89.2*	64.7
Show-o [59]	1.3B	2.0B+/665K	-	-	-	-	80.0	58.0	-	-
Liquid [57]	7B	40M / 3.5M	-	-	-	1119.3	81.1	58.4	-	42.4
ETT	1.5B	32M / 35.3M	58.8	66.6	29.3	1532.6	82.4	59.4	91.7*	56.8
ETT	3B	32M / 35.3M	61.3	67.8	32.2	1513.9	83.2	60.5	**93.3***	61.1

As shown in Table 2, our **ETT** consistently outperforms discrete counterparts, such as Chameleon [49], LWM [30], and Liquid [57], even with smaller model and data scales. This highlights the efficacy of **ETT**'s end-to-end tuning strategy, which enables to achieve strong results with fewer parameters and less data. Additionally, **ETT** achieves superior performance compared to Show-o [60] across a wide range of benchmarks, despite being trained on significantly less data. This underscores the effectiveness of **ETT**'s data utilization strategies and its ability to generalize well even with limited training resources. Furthermore, **ETT** demonstrates competitive performance against state-of-the-art (SOTA) continuous encoder-based VLMs, including QwenVL-Chat [1], EVE [7], and Janus [56], without relying on additional visual encoders for complex image encoding tasks. This not only simplifies the model architecture but also reduces computational overhead, making **ETT** a more practical and scalable solution for multimodal tasks. The success of **ETT** lies in its end-to-end training approach for visual tokenization, which seamlessly integrates multimodal understanding and generation while resolving internal conflicts within the tokenizer.

Table 3: **Comparison with existing state-of-the-art vision-language models on various text-to-image generation benchmarks**, including GenEval [14] and T2I-CompBench [16]. Note that #LLM-Param denotes the number of LLM parameters, #Data represents the training data volume, and the best results are marked in **bold**.

Method	#LLM-Param	#Data	GenEval							T2I-CompBench		
			Overall ↑	Single ↑	Two ↑	Counting ↑	Colors ↑	Position ↑	ColorAttr ↑	Color ↑	Shape ↑	Texture ↑
Generation on Continuous Features:												
SEED-X [13]	17B	-	0.51	0.96	0.65	0.31	0.80	0.18	0.14	-	-	-
PixArt-α [6]	0.6B	25M	0.48	0.98	0.50	0.44	0.80	0.08	0.07	68.86	55.82	70.44
SD v1.5 [41]	1B	2B	0.43	0.97	0.38	0.35	0.76	0.04	0.06	37.50	37.24	42.19
SD v2.1 [41]	1B	2B	0.50	0.98	0.37	0.44	**0.85**	0.07	0.17	56.94	44.95	49.82
SDXL [36]	2.6B	-	0.55	0.98	0.44	0.39	**0.85**	0.15	0.23	63.69	54.08	56.37
DALL-E2 [39]	6.5B	650M	0.52	0.94	0.66	0.49	0.77	0.10	0.19	57.50	54.64	63.74
DALL-E3 [2]	-	-	**0.67**	0.96	**0.87**	0.47	0.83	0.43	**0.45**	**81.10**	**67.50**	**80.70**
SD3 [9]	2B	-	0.62	0.98	0.74	**0.63**	0.67	0.34	0.36	-	-	-
Generation on Discrete Features:												
LlamaGen [45]	0.8B	60M	0.32	0.71	0.34	0.21	0.58	0.07	0.04	-	-	-
Show-o [59]	1.3B	35M	0.53	0.95	0.52	0.49	0.82	0.11	0.28	-	-	-
LWM [30]	7B	-	0.47	0.93	0.41	0.46	0.79	0.09	0.15	-	-	-
Chameleon [48]	34B	-	0.39	-	-	-	-	-	-	-	-	-
Emu3 [54]	8B	-	0.66	**0.99**	0.81	0.42	0.80	**0.49**	**0.45**	79.13	58.46	74.22
Janus [55]	1.3B	-	0.61	0.97	0.68	0.30	0.84	0.46	0.42	-	-	-
ETT	1.5B	30M	0.63	0.98	0.81	0.25	0.84	0.48	**0.45**	81.03	58.19	72.14

https://doi.org/10.52202/085713-1589

4.3 Visual Generation Evaluation

We comprehensively evaluate the text-to-image generation capabilities of our model against previous diffusion-based and autoregressive-based SOTA methods, including both multimodal specialists and generalists, on widely adopted benchmark datasets GenEval [14] and T2I-CompBench [16]. As demonstrated in Table 3, our method achieves competitive performance while utilizing much fewer LLM parameters and a smaller-scale training dataset. Specifically, under the inference configuration of top-k=131,072 (*i.e.*, visual vocabulary size) and top-p=1.0, our model attains an overall score of 0.63 on the GenEval dataset, outperforming advanced diffusion models such as SDXL [36]. Furthermore, our approach surpasses autoregressive-based methods, including both specialists (*e.g.*, LlamaGen [45]) and generalists (*e.g.*, Chameleon [48]), requiring either less training data or fewer model parameters. When enhanced with prompt rewriting, the achieved model accuracy by our method closely approaches the performance of current leading models such as DALL-E 3 [2] and EMU3 [54]. On T2I-CompBench dataset [16], our model achieves promising performance with scores of 81.03, 58.19, and 72.14 on color, shape, and texture pattern, respectively, demonstrating competitive performance compared to SOTA diffusion-based counterparts. These results fully prove the efficacy of our end-to-end vision tokenizer tuning approach, highlighting its ability to achieve superior multimodal understanding and generation performance across diverse benchmarks.

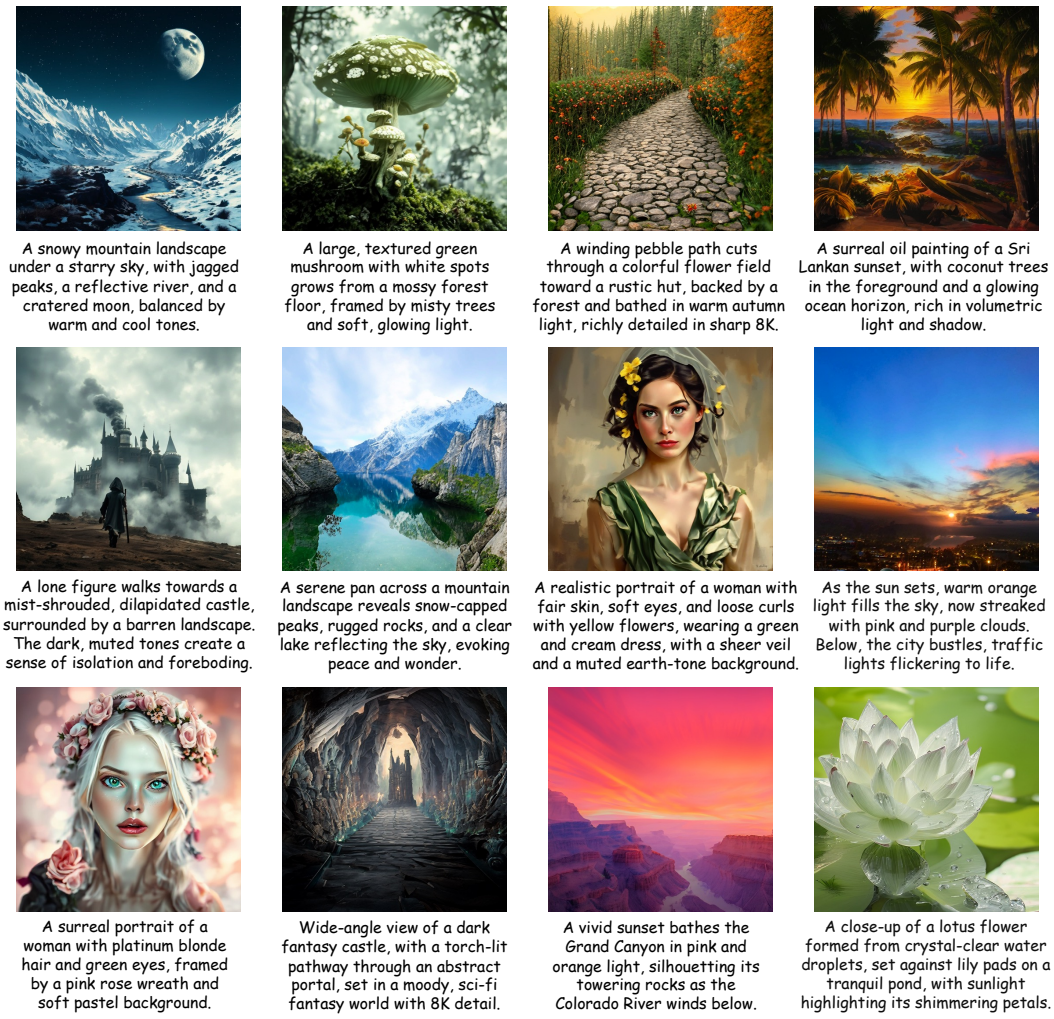

Figure 2: **Visual generation results with our ETT.** We present 512 × 512 results spanning different styles, subjects, and scenarios. Note that the presented prompts are simplified versions, which convey the general meaning.

https://doi.org/10.52202/085713-1589

Fig. 2 presents qualitative results generated by **ETT**, demonstrating its ability to follow prompts accurately while generating diverse visual content. The model exhibits proficiency in producing images across various artistic styles, subjects, and backgrounds, adapting to different compositional structures and aesthetic preferences.

4.4 Ablation Study

To verify the effectiveness of our **ETT** for downstream multimodal generation and understanding tasks, we conduct comprehensive ablation studies on several prevalent understanding benchmarks (*e.g.*, SEEDBench-Img [28], GQA [17], TextVQA [44] and MME-Perception [11]) that assess diverse capabilities, as well as the GenEval dataset [14] for the evaluation of text-to-image generation.

Table 4: **Ablation study on the benefits of ETT for multimodal understanding and generation.**

Visual	Tuning	**ETT** for Und.				**ETT** for Gen.
		$SEED^I$	GQA	TQA	MME^P	GenEval-O↑
Index	✗	54.8	50.9	40.1	1028.7	0.40
Embed	✗	54.8	52.6	40.9	1034.0	0.34
Embed	✓	**60.0**	**54.8**	**46.9**	**1124.7**	**0.43**

End-to-End Tuning Benefits. We first probe into the effectiveness of our **ETT** for promoting multimodal downstream tasks. To ensure a fair comparison in validating the potential of **ETT** in optimizing vision tokenizer's feature representations, we train all models for understanding and generation tasks with SOL-recap. Additionally, we apply LLaVA-mix-665K [31] for an extra supervised fine-tuning stage for understanding tasks. As presented in Table 4, introducing **ETT** consistently yields significant performance improvements across both understanding and generation tasks, compared with the traditional tokenizer exploitation manner. Specifically, without end-to-end tuning, replacing the discrete indices with codebook embeddings partially alleviates the issue of information loss and brings notable performance gains on multimodal understanding benchmarks. Although this replacement degrades the visual generation performance, it establishes a fully differentiable model architecture, allowing for end-to-end optimization. Building on this foundation, incorporating end-to-end tuning the visual tokenizer further enhances performance for both understanding and generation tasks compared with the conventional setting (*i.e.*, first row), particularly on tasks that heavily rely on visual features (*e.g.*, $\uparrow 5\%$ for general visual question answering [28] and $\uparrow 6\%$ for optical character recognition [44]).

Table 5: **Ablation study on the impact of ETT for the trade-offs between multimodal perception and visual reconstruction.**

Tuning Tasks for VQ	α	Understanding				Reconstruction		
		$SEED^I$	GQA	TQA	MME^P	ImageNet-rFID↓	ImageNet-PSNR↑	ImageNet-SSIM↑
-	-	54.8	52.6	40.9	1034.0	**1.03**	**21.73**	**0.60**
Und. only	-	**61.2**	**55.2**	**48.0**	**1164.3**	45.70	13.22	0.19
Und.&Rec.	0.25	60.0	54.8	46.9	1124.7	1.65	20.89	0.58
Und.&Rec.	0.5	59.9	54.7	46.9	1118.1	1.66	21.14	0.59
Und.&Rec.	1.0	59.3	53.9	45.7	1088.2	1.50	21.48	**0.60**

Trade-offs between I-to-T & Reconstruction. Then, we investigate the inherent task trade-off between visual reconstruction and multimodal understanding of **ETT**. The commonly used reconstruction metrics including rFID, PSNR and SSIM are all adopted for the comprehensive evaluations. As shown in Table 5, compared to untuned baseline (*i.e.*, first row), tuning vision tokenizer consistently delivers substantial gains for the understanding task, albeit at the cost of reconstruction performance, which deteriorates to varying extents. Specifically, tuning the vision tokenizer with only image-to-text understanding task (*i.e.*, second row) yields the best performance on various understanding benchmarks but greatly deteriorate in reconstruction, *e.g.*, the rFID on ImageNet 256×256 setting dramatically drops from 1.033 to 45.701. Introducing auxiliary reconstruction target with a small weight 0.25 slightly reduces understanding accuracy while significantly improving the reconstruction, indicating the importance of joint training on both understanding and reconstruction tasks. Increasing the reconstruction weight α to 1.0 achieves the best reconstruction accuracy but results in the weakest

https://doi.org/10.52202/085713-1589

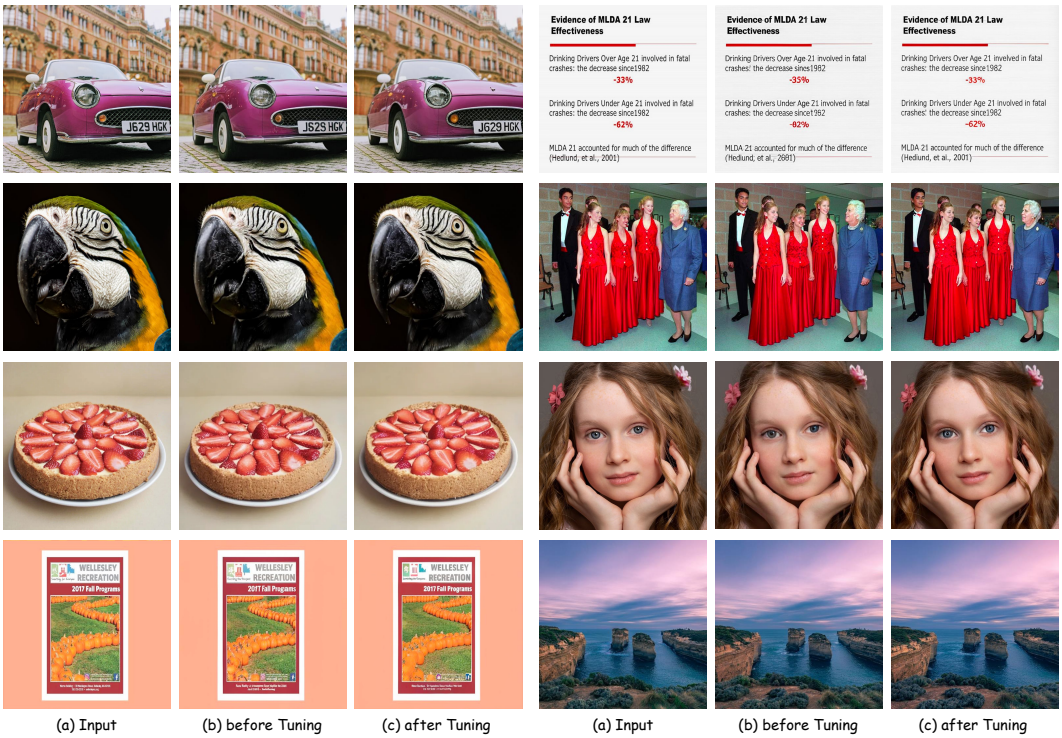

Figure 3: **Comparison of visual reconstruction results of the input image before and after end-to-end tuning by our ETT.** The vision tokenizer tuned by our **ETT** benefits from retaining the original rich low-level detail representations while being effectively injected with high-level semantics, producing visual details comparable to the pre-tuned counterpart and even performs better in some detail reconstruction, *e.g.*, text rendering.

perception capability. Therefore, to balance both understanding and reconstruction tasks, we choose 0.25 as the default reconstruction loss weight α.

Besides, we also visualize the reconstruction results before and after introducing **ETT** in Figure 3. The vision tokenizer, when tuned with **ETT**, generates visual details comparable to its untuned counterpart while enhancing specific aspects, such as text rendering. This suggests that **ETT** can not only preserve the original rich low-level details but also improve high-level semantic representations.

5 Conclusion

In this work, we focus on addressing the representation bottleneck of the vision tokenizer for multimodal learning. We introduce a simple yet effective approach of end-to-end vision tokenizer tuning, termed **ETT**. **ETT** involves codebook embeddings instead of solely discrete indices and applies token-level caption loss for end-to-end optimization of both the tokenizer and downstream training. **ETT** significantly enhances the multimodal understanding and generation with a decoder-only architecture, while almost preserving the tokenizer's reconstruction capability and even boosting reconstruction performance on specific aspects such as text rendering.

6 Limitations and Future Topics

One potential limitation of this work is that both the data scale for end-to-end fine-tuning and the model capacity could be further expanded to enhance visual representations and downstream task performance. Moreover, our current approach primarily focuses on designing a simple yet effective framework that optimizes the visual features of existing vision tokenizers, leveraging the semantic capabilities of LLMs, rather than constructing a vision tokenizer inherently designed for

https://doi.org/10.52202/085713-1589

both understanding and generation through unified end-to-end training. While **ETT** shows the potential of using LLM-driven semantic feedback to enhance vision tokenization, it still relies on fine-tuning pre-existing tokenizers rather than developing one from the ground up. Therefore, for future research we will explore end-to-end training of a vision tokenizer from scratch, aiming to create a more comprehensive and adaptable representation for multimodal tasks. Besides, extending beyond image and text modalities, such as incorporating video and audio, presents an exciting avenue for further advancements. We hope that our simple yet effective method can contribute to the broader development of multimodal foundation models beyond visual generation and understanding.

Acknowledgment

We thank all the insightful reviewers for the helpful suggestions, and the colleagues at Beijing Academy of Artificial Intelligence for support throughout this project. This research is supported by Artificial Intelligence-National Science and Technology Major Project (2023ZD0121200) and the National Natural Science Foundation of China (62437001, 62436001, U21B2043), and the Natural Science Foundation of Jiangsu Province under Grant BK20243051.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 1(2):3, 2023. 6

[2] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023. 6, 7

[3] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 5

[4] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 3

[5] 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. 3

[6] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 6

[7] Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. *arXiv preprint arXiv:2406.11832*, 2024. 4, 6

[8] Haiwen Diao, Xiaotong Li, Yufeng Cui, Yueze Wang, Haoge Deng, Ting Pan, Wenxuan Wang, Huchuan Lu, and Xinlong Wang. Evev2: Improved baselines for encoder-free vision-language models. *arXiv preprint arXiv:2502.06788*, 2025. 4, 5

[9] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 6

[10] 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. 3

[11] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. MME: A comprehensive evaluation benchmark for multimodal large language models. *arXiv: 2306.13394*, 2023. 5, 6, 8

[12] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *NeurIPS*, 36, 2024. 5

[13] Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. Seed-x: Multimodal models with unified multi-granularity comprehension and generation. *arXiv preprint arXiv:2404.14396*, 2024. 6

[14] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36, 2024. 6, 7, 8

[15] Shuhao Gu, Jialing Zhang, Siyuan Zhou, Kevin Yu, Zhaohu Xing, Liangdong Wang, Zhou Cao, Jintao Jia, Zhuoyi Zhang, Yixuan Wang, et al. Infinity-mm: Scaling multimodal performance with large-scale and high-quality instruction data. *arXiv preprint arXiv:2410.18558*, 2024. 4, 5

[16] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 2023. 6, 7

[17] Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, pages 6700–6709, 2019. 5, 6, 8

[18] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 3

 https://doi.org/10.52202/085713-1589

[19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. *arXiv: 2304.02643*, 2023. 5

[21] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023. 3

[22] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale. *arXiv: 1811.00982*, 2018. 5

[23] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 5

[24] LAION. Laion-aesthetics. https://laion.ai/blog/laion-aesthetics/, 2022. 5

[25] 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. 3

[26] Bo Li, Kaichen Zhang, Hao Zhang, Dong Guo, Renrui Zhang, Feng Li, Yuanhan Zhang, Ziwei Liu, and Chunyuan Li. Llava-next: Stronger llms supercharge multimodal capabilities in the wild, 2024. 2

[27] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 2, 4, 5

[28] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv: 2307.16125*, 2023. 5, 6, 8

[29] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *EMNLP*, pages 292–305, 2023. 5, 6

[30] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with blockwise ringattention. *arXiv preprint arXiv:2402.08268*, 2024. 6

[31] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv: 2310.03744*, 2023. 5, 6, 8

[32] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 6

[33] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? *arXiv: 2307.06281*, 2023. 5, 6

[34] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022. 5, 6

[35] Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. *arXiv preprint arXiv:2309.15505*, 2023. 3

[36] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 6, 7

[37] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. *arXiv preprint arXiv:2412.03069*, 2024. 1, 3

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 2

[39] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022. 6

[40] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021. 3

[41] 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. 6

[42] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 35:25278–25294, 2022. 5

[43] Fengyuan Shi, Zhuoyan Luo, Yixiao Ge, Yujiu Yang, Ying Shan, and Limin Wang. Taming scalable visual tokenizer for autoregressive image generation. *arXiv preprint arXiv:2412.02692*, 2024. 3

[44] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *CVPR*, 2019. 5, 6, 8

[45] 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. 3, 6, 7

[46] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14398–14409, 2024. 2

[47] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Emu: Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*, 2023. 2

[48] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 3, 4, 6, 7

[49] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 6

[50] Qwen Team. Qwen2.5: A party of foundation models, September 2024. 5

[51] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024. 6

[52] 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. 5

[53] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[54] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 1, 3, 4, 6, 7

[55] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024. 3, 6

[56] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024. 6

[57] Junfeng Wu, Yi Jiang, Chuofan Ma, Yuliang Liu, Hengshuang Zhao, Zehuan Yuan, Song Bai, and Xiang Bai. Liquid: Language models are scalable multi-modal generators. *arXiv preprint arXiv:2412.04332*, 2024. 1, 6

 https://doi.org/10.52202/085713-1589

[58] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024. 3, 6

[59] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024. 3, 6

[60] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024. 6

[61] Rongchang Xie, Chen Du, Ping Song, and Chang Liu. Muse-vl: Modeling unified vlm through semantic discrete encoding. *arXiv preprint arXiv:2411.17762*, 2024. 1, 3

[62] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. 3

[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. 3

[64] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022. 3

[65] 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. 3

[66] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv: 2308.02490*, 2023. 5, 6

[67] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 3

[68] Yue Zhao, Yuanjun Xiong, and Philipp Krähenbühl. Image and video tokenization with binary spherical quantization. *arXiv preprint arXiv:2406.07548*, 2024. 3

[69] Yue Zhao, Fuzhao Xue, Scott Reed, Linxi Fan, Yuke Zhu, Jan Kautz, Zhiding Yu, Philipp Krähenbühl, and De-An Huang. Qlip: Text-aligned visual tokenization unifies auto-regressive multimodal understanding and generation. *arXiv preprint arXiv:2502.05178*, 2025. 1, 3

[70] Chuanxia Zheng, Tung-Long Vuong, Jianfei Cai, and Dinh Phung. Movq: Modulating quantized vectors for high-fidelity image generation. *Advances in Neural Information Processing Systems*, 35:23412–23425, 2022. 3

[71] Lei Zhu, Fangyun Wei, Yanye Lu, and Dong Chen. Scaling the codebook size of vqgan to 100,000 with a utilization rate of 99%. *arXiv preprint arXiv:2406.11837*, 2024. 3

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: The abstract and introduction accurately summarize the paper's key contributions and scope.
   Guidelines:
   - The answer NA 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 NA 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: We discuss the limitations of the work in Sec. 6.
   Guidelines:
   - The answer NA 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: [NA]
   Justification: This paper does not include theoretical results.
   Guidelines:
   - The answer NA 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.

https://doi.org/10.52202/085713-1589

- 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: The paper fully discloses the necessary information for reproducing the main experimental results. Detailed experimental settings are provided in Sec. 4.
   Guidelines:
   - The answer NA 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: [Yes]
   Justification: We use publicly accessible datasets for the main experiments in this paper. Once the blind review period is finished, we'll open-source the codes, instructions, and model checkpoints.
   Guidelines:
   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.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://nips.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, etc.) necessary to understand the results?
   Answer: [Yes]
   Justification: The paper specifies all the necessary training and test details in the experimental section, ensuring a clear understanding of the results.
   Guidelines:
   - The answer NA 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: The paper presents the results from a single run for each experiment, which is consistent with previous works on the same task.
   Guidelines:
   - The answer NA 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]

https://doi.org/10.52202/085713-1589

Justification: The paper provides detailed information on the computer resources used for each experiment within Sec. 4.

Guidelines:

- The answer NA 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: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA 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: [NA]

    Justification: This work proposes a method for end-to-end tuning of a vision tokenizer with large language models. As the contribution is methodological and does not directly target any downstream application, we believe it does not raise any immediate or foreseeable societal impacts requiring discussion.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

Justification: We do not foresee any high risk for misuse of this work.
Guidelines:

- The answer NA 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: We properly credit datasets that we use in experiments and ensure that they are properly licensed.
Guidelines:

- The answer NA 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: [NA]
Justification: The paper does not release new assets.
Guidelines:

- The answer NA 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: [NA]
Justification: The paper does not involve crowdsourcing nor research with human subjects.
Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

https://doi.org/10.52202/085713-1589

- 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: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA 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 rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The core method in our paper incorporates the powerful LLM as a central component for vision tokenizer tuning and multimodal reasoning, which significantly contributes to the originality and performance of the approach.

    Guidelines:

    - The answer NA 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 (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.