dRAE: Representation Autoencoder with Hyper-Spherical Codes

Tianren Ma[1,4] **Lin Long**[2,4] **Chuyan Chen**[3,4] **Mu Zhang**[1]
Junbo Zhao[2,4*] **Tong Zhang**[1] **Qixiang Ye**[1*]

[1] University of Chinese Academy of Sciences [2] Zhejiang University
[3] Peking University [4] Ant Group

Abstract

In this work, we aim to discretize the high-dimensional visual representations to bridge the gap with language models — a non-trivial challenge, as existing quantization methods suffer from codebook collapse, failing to scale while preserving semantic coherence. We identify the root cause as metric mismatch: standard Euclidean codebook objectives are fundamentally misaligned with the anisotropic geometry of representation space, leading to codebook embeddings with high-variance magnitude scales and uneven angular distributions that hinder scalability. To address this, we propose **Hyper-Spherical Quantization (HSQ)**, which decouples semantic content from feature magnitude via angular routing, preventing code assignment from being dominated by scale rather than meaning. The resulting discrete Representation Autoencoder (dRAE) achieves high-fidelity reconstruction while preserving semantic integrity and supporting scalable codebook budget. Extensive experiments demonstrate consistent performance gains as the vocabulary size scales to 131,072, along with 100% codebook utilization, simplified training pipeline, and strong performance across understanding and generation tasks. Project page: `https://drae-hsq.github.io`.

1 Introduction

The trajectories of visual understanding and generation have conventionally been separated by their representation designs. Visual encoders optimized for understanding [40, 35, 53], excel at mapping images to high-dimensional, semantically rich spaces via vision transformers (ViTs) [11, 18], whereas generative frameworks typically rely on variational autoencoders (VAEs) [22, 41, 13, 36] to compress images into low-dimensional manifolds. While modern VAEs are highly optimized for localized pixel reconstruction, scaling their latent dimensions is constrained by the KL-divergence prior and practical difficulties [63]. Consequently, most *unified* multimodal models resort to disjointed pipelines—employing a ViT for comprehension and an independent VAE for generation.

Recently, a promising trend has emerged to bridge this architectural gap. Frameworks [59, 68] such as the representation autoencoder (RAE) demonstrate that high-dimensional features from contrastive or self-supervised models can be directly used to reconstruct images via a simple ViT. As these pretrained representations inherently capture global semantic alignment, visual tokens derived from this space carry significantly higher generalizable information than that from a pure reconstruction VAE. This high semantic bandwidth is desirable for multimodal models, as it minimizes the need for complex cross-modal alignment [52, 51], offering a more unified foundation for integrated understanding and generation.

However, transporting such signals via discrete indices presents a significant optimization challenge. Existing vector quantization (VQ) methods [54], Fig. 1 *a*, which are designed for low-dimensional autoencoder bottlenecks, typically result in severe codebook collapse when applied on high-dimensional

*Corresponding authors. ✉`j.zhao@zju.edu.cn` ✉`qxye@ucas.ac.cn`

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

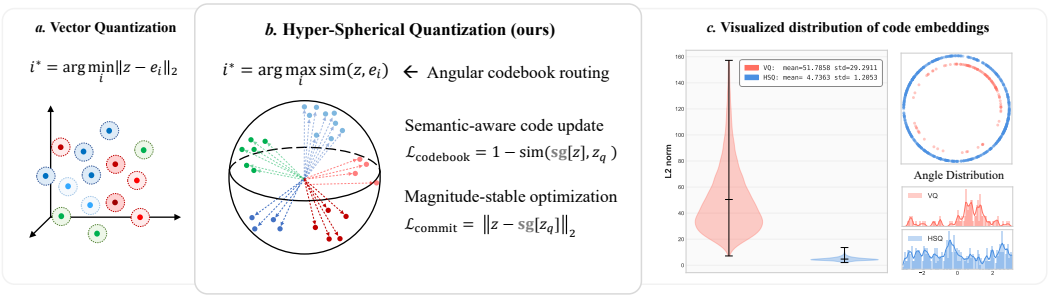

Figure 1: Illustration of the code space defined by VQ (a) and HSQ (b), and distribution of their learned codebook embeddings (c). $\text{Sim}(\cdot)$ for cosine similarity, and $\text{sg}(\cdot)$ for stop-gradient operation. For the angular analysis in c, we sample features from the central region with similar magnitude ranges and visualize them with PCA dimension reduction [56].

features (see Fig. 2). Empirical observations in recent baselines [12] reveal that the tokenizer plateaus when scaled to a vocabulary larger than 16K codes. This phenomenon neutralizes the primary advantage of representation-based designs, as the constrained vocabulary cannot adequately capture the diverse semantic capacity of the latent space.

In this study, we argue that the observed codebook collapse is largely attributable to the fundamental **metric mismatch**. While semantics in vision foundation models are predominantly encoded in the orientation of latent vectors [56], VQ-based frameworks rely on Euclidean distance for codebook assignment. This inconsistency makes the objective sensitive to the anisotropic scale of pretrained spaces, resulting in codebook embeddings with high-variance magnitude and clustered angular distributions, as evidenced in Fig. 1 c. Such magnitude-induced bias allows latent vectors with disparate norms to disproportionately dominate the tessellation, enabling certain codebook entries to "hijack" assignments based on numerical magnitude alone, regardless of their semantic alignment with the input. At the same time, simply imposing a unit-sphere prior on inputs, which means discarding the magnitude, would deteriorate reconstruction, as this operation is non-uniform across tokens, leading to structural and textural nuances lost.

To resolve this dilemma, we propose the discrete Representation Autoencoder (dRAE) powered by Hyper-Spherical Quantization (HSQ). The core innovation lies in explicitly decoupling the metric used for codebook routing from the quantization objective. By shifting from Euclidean distance to angular similarity, HSQ assigns codes based on semantic orientation while preserving essential magnitude information for the decoder. This approach significantly improves tokenization quality through two primary mechanisms: **magnitude homogenization** and **angular uniformity**. As illustrated in Fig. 1 c, HSQ constrains the codebook to a significantly more stable magnitude scale, effectively eliminating the high-variance outliers that trigger codebook collapse in vanilla VQ or SimVQ [69]. By reducing the magnitude-induced noise, the latent space attains a more isotropic, uniform distribution on the hyper-sphere. This ensures that the codebook utilizes its full capacity to represent distinct semantic directions rather than being dominated by arbitrary feature scales. Consequently, dRAE preserves the fine structural details necessary for high-fidelity synthesis while maintaining the semantic integrity required for large-scale multimodal understanding.

Through extensive experiments, we demonstrate that the proposed HSQ not only shows continuous gain on reconstruction by scaling vocabulary up to $131,072$, but also keeps high semantic fidelity for image understanding tasks. We also validate the practical efficacy of dRAE by integrating it with a generative pipeline for text-to-image generation, confirming its viability as a high-throughput tokenizer for both visual understanding and generation tasks.

2 Related Work

Visual Semantic and Generative Latent. The evolution of self-supervised and contrastive learning has established a robust foundation for multimodal models. By utilizing objectives such as masked modeling [18], self-distillation [6, 35] and image-text alignment [40, 53], visual models embed image patches into high-dimensional latents (*e.g.* $d \in [768, 1536]$), and learn semantically structured

representation that generalizes across visual understanding tasks. On the other hand, state-of-the-art generative models still operate on heavily compressed manifolds ($d \in [8, 64]$), *i.e.*, diffusion is typically built in reconstruction-trained VAE space [41, 63]. This creates a dimensional incompatibility between the semantic-rich embeddings and the compact generative latents.

Unifying Comprehension and Generation. One direct solution of aligning these disparate paths is to explore an ensemble of semantic and reconstruction objectives. VTP [62] explores end-to-end pretraining to learn representations that can unify comprehension with generation from scratch. However, it imposes extremely high demand on data collection and training budgets. A more promising trend is to utilize a pretrained encoder for more downstream tasks. VILA-U [59] uses residual quantization to align image latents with text embeddings, while TokLIP [28] bakes CLIP features into the pretrained VQGAN tokenizer through distillation and contrastive training.

Recently, RAE [68] proposes that the high-dimensional features, optimized for semantics, can be directly decoded for image reconstruction. It also shows that high-dimensional vectors, which jointly capture high semantic density and fine-rained structural, are of great potential in visual world modeling [51]. Its discrete counterpart, VQRAE [12], uses SimVQ [69] to quantize the high-dimensional tokens, but is prone to codebook collapse and of limited scaling potential. In this study, we attempt to address these limitations by introducing a scalable quantization method that stabilizes training while preserving semantic depth.

3 Preliminary

Image Reconstruction. This requires the model to learn a mapping from the input image $x \in [0, 1]^{h \times w \times 3}$ to latents and reconstruct them back to pixel space. Formally, an encoder E maps x to latent features $Z := E(x) = \{z_n\}_{n=1}^{N} \in \mathbb{R}^{N \times d}$, where $N = hw/p^2$ denotes the number of visual tokens. We will also use $z \in \mathbb{R}^d$ to denote an un-quantized embedding. A decoder D reconstructs the image $\hat{x} = D(Z)$. The model is trained to minimize a objective that encourages both pixel-level fidelity (L_1 distance) and perceptual quality (perceptual $\mathcal{L}_{\text{perc}}$ and discriminator loss $\mathcal{L}_{\text{disc}}$):

$$\mathcal{L}_{\text{rec}} = \mathrm{L}_1(\hat{x}, x) + \omega_{\text{p}}\mathcal{L}_{\text{perc}}(\hat{x}, x) + \omega_{\text{d}}\mathcal{L}_{\text{disc}}(\hat{x}, x). \tag{1}$$

Representation Autoencoder. Conventional approaches such as VAEs compress images into low-dimensional latent spaces, $i.e.$, the bottleneck, which facilitates convergence but limits the capacity to encode semantic information. In contrast, RAE removes the bottleneck design, and instead leverages high-dimensional, semantically structured embeddings directly from pretrained Vision Foundation Models (VFMs) as its latents. Specifically, RAE employs a frozen VFM as E and a trainable ViT as D. By avoiding aggressive compression, RAE maintains features that are beneficial for downstream visual understanding and generation tasks.

Vector-Quantizing Semantics. VFM features can be quantized for discrete modeling. Let $Q(\cdot)$ denote the quantization operation, the quantized features as $z_q := Q(z)$ and $Z_q := Q(Z)$. VQRAE projected the image features to 1536 dimensions, and each vector z_n is mapped to its nearest entry from a learnable codebook $\mathcal{C} = \{c_i\}_{i=1}^{K}$, where K is the codebook capacity:

$$z_q = \arg\min_{c_i \in \mathcal{C}} \|z - c_i\|_2. \tag{2}$$

The quantization objective consists of a codebook loss and a commitment loss:

$$\mathcal{L}_{\text{VQ}} = \underbrace{\|\text{sg}[z] - z_q\|_2^2}_{\mathcal{L}_{\text{codebook}}} + \beta \underbrace{\|z - \text{sg}[z_q]\|_2^2}_{\mathcal{L}_{\text{commit}}}, \tag{3}$$

where $\text{sg}[\cdot]$ denotes the stop-gradient operator and β controls the commitment strength.

To preserve the semantic structure while improving reconstruction fidelity, VQRAE also adopts a self-distillation strategy. A teacher model T (initialized from the original VFM) provides supervision to the unfrozen D, encouraging the learned features Z_I to remain close to the teacher representations:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{VQ}} + \lambda\|Z - T(x)\|_2^2. \tag{4}$$

Additionally, to improve codebook utilization, a learnable projection matrix $W \in \mathbb{R}^{d \times d}$ is introduced by SimVQ [69] to transform codebook entries as Wc_i, allowing for a more flexible alignment.

Table 1: **Diagnostic Analysis.** Comparison of understanding (a) and reconstruction (b) performance between raw and normalized image features as inputs.

(a) Multimodal Understanding

Variant	MMBench ↑	TextVQA ↑	POPE ↑
Raw Z	82.2	61.3	85.2
Norm Z'	81.1	60.7	84.3

(b) Image Reconstruction Fidelity

Variant	PSNR ↑	SSIM ↑	rFID ↓
Raw Z	22.5	0.62	4.62
Norm Z'	20.6	0.51	9.57

4 Methodology

4.1 Anisotropy in High-dimensional Space

In high-dimensional Euclidean space, a standard Gaussian distribution exhibits the *thin-shell* concentration property [55, 23]: if $z \sim \mathcal{N}(0, I_d)$, its norm $\|z\|_2$ concentrates sharply around $\sqrt{d}$ as $d \to \infty$. Vision encoders such as CLIP [40] are trained with contrastive objectives that encourage features to spread uniformly across directions-a pressure analogous to the isotropic structure of a Gaussian [5]. Related self-supervised approaches like DINO [6, 35] series and JEPA [1] also yield highly regular representations [4], which have been shown to encode density-related structure that can be exploited with Gaussian models. Their patch embeddings, as suggested by recent studies [24, 56], similarly concentrate on a thin spherical shell, with meaningful information encoded in both the direction (angular) and the radius (magnitude) of each feature vector. We conduct diagnostic experiments to empirically investigate their function, with the aim of validating two hypotheses:

H1. *Semantic information roughly lies on a hyper-sphere for understanding.* We train two variants of an MLLM following LLaVA's ViT-MLP-LLM architecture [30]. The original version receives raw image features Z as MLP's inputs. In the normalized version, we apply ℓ_2 normalization such that all visual features $Z' = Z/\|Z\|_2$ reside on a unit hyper-sphere. As shown in Tab. 1a, the performance gap across diverse benchmarks—ranging from visual perception (MMBench [31]) to hallucination (POPE [27]) —is negligible ($\sim 1\%$). This confirms that the semantic signal is almost entirely preserved in the feature's directional orientation for high-level understanding.

H2. *Magnitude is structurally essential for image reconstruction.* Following RAE, we train a ViT decoder to reconstruct pixels from either Z or Z'. As shown in Tab. 1b, models trained on normalized features struggle to recover global structure, resulting in worse performance on reconstruction metrics.

The diagnostic results suggest that, to build a tokenizer that captures both high semantic density and fine-grained structural fidelity, we need a scalable quantization algorithm that preserves both the angular and magnitude information of high-dimensional features.

4.2 Hyper-Spherical Quantization

The Unreliable Euclidean. As discussed above, representation features approximately concentrate on a hyper-spherical shell rather than a full Euclidean space. However, VQ optimizes codebooks using the ℓ_2 distance, as in Eq. 3. Such an objective is linearly sensitive to variations in feature norms, which biases the partitioning toward radial differences rather than angular structure. Consequently, samples with larger or varying norms can disproportionately influence the tessellation, irrespective of their directional alignment, as shown in Fig. 1. We posit that this mismatch is a primary cause of the codebook collapse phenomenon observed in Fig. 2.

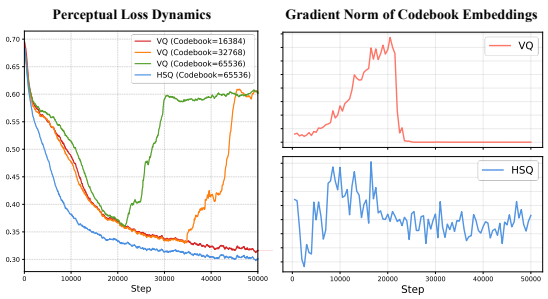

Figure 2: Left: Training curve under identical settings. Right: Gradient norm of codebook during training, where VQ's collapse mode can be observed.

Algorithm 1 Classic Vector Quantization

Input: $Z \in \mathbb{R}^{N \times d}, C, \beta$
$D_{i,j} \leftarrow \|Z_i - C_j\|_2^2$
$I_i \leftarrow \arg\min_j D_{i,j}$
$Z_q \leftarrow C[I]$
$\mathcal{L}_{\text{codebook}} \leftarrow \|\text{sg}[Z] - Z_q\|_2^2$
$\mathcal{L}_{\text{commit}} \leftarrow \|Z - \text{sg}[Z_q]\|_2^2$
$\mathcal{L}_{\text{VQ}} \leftarrow \mathcal{L}_{\text{codebook}} + \beta \mathcal{L}_{\text{commit}}$
Return: $\mathcal{L}_{\text{VQ}}$

Hyper-Spherical Quantization

```
# Compute cosine similarity
```
3: $S_{i,j} \leftarrow \frac{Z_i}{\|Z_i\|_2} \cdot \frac{C_j}{\|C_j\|_2}$

```
# Angular routing
```
4: $I_i \leftarrow \arg\max_j S_{i,j}$
5: $Z_q \leftarrow C[I]$

```
# Spherical codebook loss
```
6: $\mathcal{L}_{\text{codebook}} \leftarrow 1 - \text{sg}[\frac{Z}{\|Z\|_2}] \cdot \frac{Z_q}{\|Z_q\|_2}$

Objective Decoupling. To mitigate these challenges, we propose Hyper-Spherical Quantization (HSQ). Our core insight is to decouple the semantic orientation of a feature from its magnitude during the quantization step. Following VQRAE, we add a learnable linear projection on the raw image features, then, HSQ replaces the standard Euclidean assignment with angular routing:

$$I_i = \arg\max_j \frac{Z_i \cdot C_j}{\|Z_i\|_2 \|C_j\|_2}. \tag{5}$$

By transitioning to an angular-based metric, we assign the codebook based on purely the directional relationships learned during pretraining. To maintain this constraint, we also rewrite the codebook loss $\mathcal{L}_{\text{codebook}}$ as a spherical objective, which constrains codebook updates to the tangent space.

$$\mathcal{L}_{\text{codebook}} = 1 - \text{sg}\left[\frac{Z}{\|Z\|_2}\right] \cdot \frac{Z_q}{\|Z_q\|_2}. \tag{6}$$

How is the magnitude optimized? The magnitude scaling of code vectors is not arbitrary. In fact, replacing all objectives with angular-based methods leads to unstable and slow convergence, since the magnitude is completely unsupervised (see Sec. 6 for details). Therefore, we retain the Euclidean objective for commitment loss deliberately, as it provides magnitude-aware guidance to the projected features. The core comparison is shown above. Please refer to Append. A for the complete training algorithm, where full implementation like the straight-through estimator (STE) and codebook projection are included.

4.3 Distinction from Existing Quantization Schemes

To fully appreciate our design, it is instructive to compare HSQ with two popularized quantization paradigms: Finite Scalar Quantization (FSQ) [34] and Spherical Vector Quantization (SVQ) [66].

FSQ dispenses with a learnable codebook entirely, instead bounding the continuous latents via a squashing function (*e.g.*, $\tanh$) and rounding them to a predefined discrete grid. While FSQ effectively eliminates collapse by removing competitive codebook learning, it is confined to low dimensions. The implicit vocabulary size in FSQ scales exponentially with the dimension d as $K = L^d$, where L is the number of bins per dimension. This will yield *intractable* vocabulary size for high-dimensional features. In contrast, HSQ operates directly in the native high-dimensional space while maintaining a highly utilized codebook.

SVQ projects latents onto a unit sphere, but it traditionally do so to impose a uniform yet strict prior on the latent space for training generative models from scratch. Consequently,it passes the *normalized* vectors to the decoder. Crucially, HSQ differs by explicitly decoupling the angular routing from the magnitude-aware reconstruction, which does not enforce SVQ's unit sphere prior on the inputs. As detailed in Sec. 4.2, we only use the spherical geometry during the nearest-neighbor assignment. For the actual representation passed to the decoder (and used in the commitment loss), we retrieve the *unnormalized* code vector.

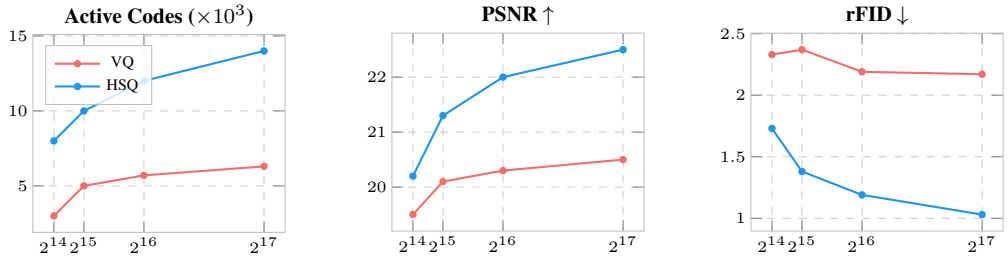

Figure 3: **Scaling behavior with $K = 2^{14} \sim 2^{17}$ vocabulary size.** From left to right: active code per batch, reconstruction PSNR, and rFID.

5 Experiment

5.1 Setups

We provide a brief overview of the implementation in this section. Full experimental details and hyperparameters are included in Sec. A. Throughout the experimental section, VQ denotes vector quantization trained with the conventional ℓ_2 objective. Unless specified, all VQ implementations follow the SimVQ design, where the codes are projected with an linear projection layer.

Implementation. Following VQRAE, we utilize a pretrained SigLIP2 ViT-So400M [53] as the image encoder, and use a ViT of symmetric design for the decoder. Unlike previous methods [12, 28] that may use two-stage training for stability and performance trade-off, dRAE is trained end-to-end with distillation loss in one stage. With no specific embedding initialization, anti-collapse trick nor curriculum learning, the training pipeline is much simplified.

For image understanding, we employ Qwen2.5-7B [3] as the LLM backbone, and adapt the commonly used visual instruction tuning pipeline as in LLaVA-1.5 [30]. For text-to-image (T2I) generation, we use Flan-T5-XXL [8] for text conditioning, and a diffusion transformer with discrete prediction head [33, 42]. The generative training follows the discrete diffusion paradigm, where images are tokenized into discrete indices and masked randomly during training. The training process is optimized via a time-weighted masked cross-entropy loss. We also train several class-to-image (C2I) generative variants on ImageNet for ablative study. See A.3 for details.

Evaluation Metrics. We evaluate reconstruction quality using reconstruction FID (rFID) [19], Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM) on the ImageNet-1K [9] validation set. For multimodal understanding, we evaluate on GQA [21], TextVQA (TQA) [45], MMBench-en (MMB) [31], MME-Perception (MME-P) [14], and SEEDBench-Img (SEED) [26]. For C2I generation, we evaluate using generation FID (gFID) and Inception Score (IS) [43]. For T2I generation, we evaluate on GenEval [16] and DPG-Bench [20].

5.2 Image Reconstruction

Unlike prior observations that suggest limited gains from increasing vocabulary size [12], the proposed approach demonstrates consistent scaling behavior. As shown in Fig. 3, increasing the vocabulary size leads to higher code utilization, which correlates with improved reconstruction fidelity. As a contrast, the VQRAE (VQ) baseline demonstrates marginal improvement of active code number and PSNR when scaling up the vocabulary size. Note that the active code number is calculated by summing up the unique code usage per batch during training.

Our hyper-spherical code design exhibits strong resistance to codebook collapse. Under a vocabulary size of $K = 65{,}536$ with batch size 64 per GPU, the VQRAE baseline only activates $\sim$6K codes per forward pass, whereas our method consistently more than $\sim$12K codes, while keeping a high global code utilization ratio ($> 90\%$) without stochastic sampling tricks. For HSQ, larger vocabularies also yield consistently better reconstruction performance, while VQ method's improvement (before collapse) is relatively marginal. As can be seen in Tab 2, under the same codebook budget $(16,384)$, dRAE outperforms VQRAE with significant margins. With a $131,072$ codebook, dRAE achieves 0.42 rFID, and its PSNR and SSIM are also the best among the compared discrete tokenizers.

Table 2: **Evaluation on reconstruction quality.** NA refers to direct utilization of pretrained features. Dim. indicates latent dimension (bottleneck) for autoencoders. *: See A.5 for evaluation details.

Autoencoders	Method	Codebook	Dim.	rFID ↓	PSNR ↑	SSIM ↑
Low-dimensional						
SD-VAE [41]	VAE	–	32	0.49	26.10	0.72
UniLIP* [49]	VAE	–	32	0.79	22.99	0.67
MUSE-VL [61]	VQ	32768	8	2.26	20.14	0.65
QLIP [67]	BSQ	2^{28}	28	3.21	23.16	0.63
Representational						
RAE [68]	NA	–	768	0.62	19.20	0.44
VILA-U [59]	RQ	16384	1152	1.25	–	–
VQRAE* [12]	VQ	16384	1152	1.31	22.33	0.67
dRAE	HSQ	16384	1152	**0.69**	**24.03**	**0.70**
dRAE	HSQ	131072	1152	**0.42**	**24.52**	**0.72**

Table 3: **Evaluation on multimodal understanding**. Res. for image resolution. All the models listed use $\sim$ 7B LLM as the backbone. Top-two best results are highlighted.

Method	**Vision Encoder**	**Res.**	**GQA**	**TQA**	**MMB**	**MME-P**	**SEED**
w/ Understanding Only							
LLaVA-1.5 [30]	CLIP ViT-L	336	62.0	46.1	64.3	1510.7	58.6
LLaVA-OV [25]	SigLIP ViT-So	384	–	–	80.8	1580.0	75.4
w/ Unified Tokenizer							
VILA-U [59]	SigLIP ViT-So	384	60.8	60.8	-	1401.8	59.0
UniTok [32]	Vitamin-L	256	-	-	-	1448.0	-
QLIP [67]	CLIP ViT-L	392	61.8	55.2	-	1498.3	-
TokLIP [28]	SigLIP2 ViT-So	384	59.5	-	67.6	1488.4	70.4
Tar [17]	SigLIP2 ViT-So	384	61.3	-	74.4	**1571.0**	**72.1**
VQRAE [12]	SigLIP2 ViT-So	512	**63.6**	58.8	67.6	1494.2	62.8
dRAE	SigLIP2 ViT-So	384	62.1	**63.8**	**81.4**	1468.8	71.1
dRAE	SigLIP2 ViT-So	512	**62.4**	**67.7**	**81.5**	**1535.1**	**72.7**

5.3 Multimodal Tasks

Image Understanding. dRAE demonstrates competitive performance on multimodal understanding benchmarks. We follow VQRAE to evaluate the tokenizer's performance using its tuned image encoder. Note that VQRAE is trained with original encoder but tested with the tuned one, while our method uses the encoder optimized for reconstruction throughout MLLM training and testing, consistent with other baselines. As shown in Tab. 3, while the tokenizer training is end-to-end, dRAE maximally preserved the semantic knowledge encoded in the pretrained encoder.

Visual Generation. We evaluate the T2I generation performance of dRAE in Tab. 4. Despite its lightweight architecture ($\sim$700M trainable parameters) and relatively modest training scale, our generative model demonstrates competitive capabilities against baselines trained on significantly larger datasets. This underscores that semantic latents constructed from VFMs serve as a powerful and highly efficient foundation for generative downstream tasks. For C2I generation, models utilizing HSQ-derived tokenizers consistently outperform their VQ-derived counterparts across all codebook sizes, yielding substantial improvements in both gFID and Inception Score (IS), as detailed in Tab. 5. Furthermore, we integrate iREPA [46] to enhance internal representation alignment. As shown in Tab. 6, the HSQ-based approach benefits more pronouncedly from this alignment strategy. Specifically, while iREPA accelerates early convergence for both tokenizers, the VQ-based baseline suffers from an early performance plateau and noticeable training oscillations.

6 Ablative Study

Loss Design. We evaluate the contribution of each component in our design by systematically ablating on the routing mechanism and loss formulations. Table 7 summarizes the evaluated variants. The

Table 4: **Evaluation on text-to-image generation**. Data refers to the image-text pairs traversed during training. We leave the exhaustive data and model curation for future study.

Model	GenEval							DPG-Bench	Data
	Single	Two	Count	Color	Pos.	Attr.	Overall	Overall	
Continuous Generation									
SDv1.5 [41]	0.97	0.38	0.35	0.76	0.04	0.06	0.43	63.18	$>$ 1B
DALL-E 3 [29]	0.96	0.87	0.47	0.83	0.43	0.45	0.67	83.50	$>$ 1B
Discrete Generation									
LlamaGen [48]	0.71	0.34	0.21	0.58	0.07	0.04	0.32	64.84	60M
TokenFlow [38]	–	–	–	–	–	–	0.55	73.38	60M
Show-o [60]	0.95	0.52	0.49	0.82	0.11	0.28	0.53	–	35M
Meissonic [2]	0.99	0.66	0.42	0.86	0.10	0.22	0.54	–	$>$200M
Janus [58]	0.97	0.68	0.30	0.84	0.46	0.42	0.61	79.68	$>$100M
dRAE	0.98	0.69	0.45	0.92	0.30	0.39	**0.63**	**80.58**	**12M**

Table 5: Ablation on C2I generation. Using simplest sampler without hyperparameter searching.

Method	Codebook	rFID	gFID↓	IS↑
VQ	32768	2.25	5.37	251.3
VQ	65536	2.21	5.51	264.3
HSQ	32768	2.20	4.83	268.4
HSQ	65536	2.14	4.45	287.3

Table 6: Ablation on alignment. iREPA for improved representation alignment [46].

Method	Configuration	gFID↓	IS↑
VQ	w/o iREPA	6.89	215.3
VQ	w/ iREPA	7.16	226.1
HSQ	w/o iREPA	6.65	251.4
HSQ	w/ iREPA	6.11	256.3

Table 7: Ablation of objective metric on reconstruction tasks. ℓ_2 for standard Eucildean metric, and θ for cosine-similarity based method.

Assign	$\mathcal{L}_{\text{codebook}}$	$\mathcal{L}_{\text{commit}}$	rFID	SSIM
ℓ_2	ℓ_2	ℓ_2	3.59	0.62
θ	ℓ_2	ℓ_2	3.31	0.62
θ	θ	ℓ_2	**3.02**	**0.65**
θ	θ	θ	12.7	0.33

Table 8: Ablation of tokenizer design on multimodal understanding tasks under identical visual instruction tuning setting.

Method	Codebook	GQA	MMB
VQ	16384	34.1	43.8
VQ	65536	33.8	42.0
HSQ	16384	34.3	44.1
HSQ	65536	35.8	45.6

baseline (line 1) uses Euclidean routing with ℓ_2-based codebook and commitment loss. Replacing Euclidean distance with cosine similarity (line 2) already improves performance, indicating the benefit of directional matching in the latent space. Building on this, our method (line 3) introduces a cosine-based $\mathcal{L}_{\text{codebook}}$ while retaining an ℓ_2 commitment loss. The full-spherical variant (line 4) applies cosine metrics to all components.

The results show that the best performance is achieved by combining a cosine-based $\mathcal{L}_{\text{codebook}}$ with an ℓ_2 commitment loss. This allows the codebook to align with the directional structure of the latent space, while the Euclidean commitment term preserves feature magnitude and guides the projection to meet with codebook's distribution. When both losses are defined in cosine space, performance degrades significantly, suggesting that magnitude information remains important for convergence. We also provide additional experiment results with DINOv2 as the encoder in A.2.

Impact on Semantics. We further explore the use of quantized features for training multimodal models and evaluate the impact of different codebook designs on downstream understanding performance. As shown in Tab 8, performance with HSQ consistently improves as the codebook size increases, whereas VQ-based methods exhibit relatively stagnant behavior. This observation further supports our claim that semantically aligned encoding leads to superior performance.In practice, we also observe that VQ-based approaches suffer from notable instability during training: the reconstruction loss exhibits persistent oscillations, while the distillation loss tends to undergo an irreversible increase in the later stages of training, indicating a growing deviation from the pretrained teacher model. Such issues are not observed with HSQ.

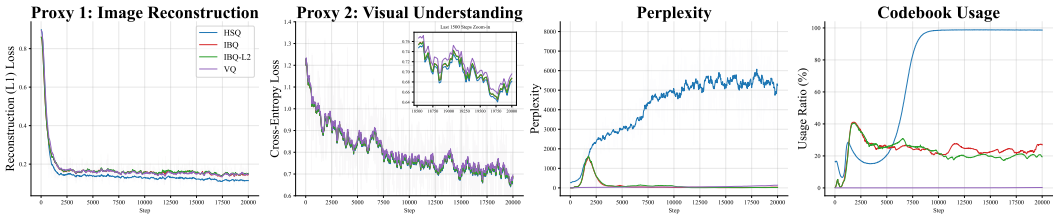

Figure 4: Comparison of quantization methods under the joint image-semantic optimization.

7 Beyond Image Reconstruction

The main experiments in this paper follow the RAE paradigm, where image reconstruction serves as a *proxy task* for training the tokenizer. Nevertheless, the core idea of HSQ is to learn an effective mechanism for quantizing high-dimensional representations, which should be applicable regardless of the choice of proxy task. Several recent works [15, 10, 50, 57, 37] have discussed related directions. We explore alternative proxy and report two preliminary experiments:

Feature Reconstruction. With the encoder E frozen, we get the encoded image features $z = E(x)$ and train a codebook together with a ViT decoder to reconstruct the features $\hat{z}$. The reconstruction objective is defined as $\mathcal{L}_{\text{rec}} = \text{L}_2(\hat{z}, z)$. We train the quantizer and decoder for 100K steps and evaluate the PSNR and cosine similarity between z and $\hat{z}$. As shown in Tab. 9, HSQ significantly outperforms VQ. Nevertheless, reconstructing high-dimensional patch embeddings is more challenging than reconstructing pixel values, which we attribute to the intrinsic noise in the encoder representations and the absence of perceptual supervision [65], from which VAEs benefit.

Table 9: Comparison of VQ and HSQ on feature reconstruction.

Method	PSNR ↑	Sim. ↑
VQ	4.91	0.81
HSQ	**8.24**	**0.92**

Joint Image-Semantic Modeling. We insert the quantizer between the frozen visual encoder E and the trainable language backbone D_{LLM} of Qwen-3.5 [39], while attaching a trainable ViT decoder D_{pixel} in parallel. The codebook is then trained end-to-end using both image reconstruction and image understanding objectives, allowing supervision to flow through a unified training objective. Let z_q denote the quantized image features, the reconstructed image $\hat{x} = D_{\text{pixel}}(z)$, and y as the language response for query q about image x. We use 4M image-query-response pairs $\{x, q, y\}$ from LLaVA-OneVision [25] as the training dataset. The overall objective combines the language modeling loss and the image reconstruction loss as $\mathcal{L}_{\text{rec}} = -\sum_t \log p_{D_{\text{LLM}}}(y_t \mid y_{<t}, q, z_q) + \text{L1}_{D_{\text{pixel}}}(\hat{x}, x)$.

Through experiments, we found that an entropy regularization loss [7, 64] is essential for effective codebook utilization when an LLM is used as the decoder. Since this loss is already included in the IBQ family, we plug it to HSQ and VQ for a fair comparison. We compare HSQ with VQ, IBQ [44], and the ℓ_2-normalized variant of IBQ [15], with a vocabulary size of $K = 131{,}072$. The entropy regularization uses a temperature-scaled softmax with temperature $\tau = 0.01$ for all methods.

As shown in Fig. 4, VQ performs the worst among all compared methods, exhibiting slow convergence and poor codebook usage. The IBQ variants benefit from stochastic sampling and consequently achieve higher code utilization. HSQ, despite not relying on randomized sampling, achieves the highest codebook utilization and the fastest convergence in both image reconstruction and understanding tasks by simply adopting angular-based routing and codebook updates. These results further demonstrate the effectiveness of HSQ for quantizing high-dimensional representations.

8 Conclusion

We presented dRAE, a Representation Autoencoder with Hyper-Spherical Codes, directly tackling the pervasive codebook collapse issue in high-dimensional discrete visual tokenizers. By aligning the codebook update dynamics more closely with the underlying semantic latent distribution, our approach achieves exceptionally high codebook utilization, fast convergence and robust reconstruction quality with minimal anti-collasping tricks. This study provides a fresh insight for training visual tokenizers, as well as laying a scalable foundation for training next-generation MLLMs.

9 Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC) under Grant 62225208, 62650006, 62521007 and CAS Project for Young Scientists in Basic Research under Grant No.YSBR-117. This work was also supported by Ant Group Research Intern Program.

References

[1] 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. *arXiv preprint arXiv:2301.08243*, 2023.

[2] Jinbin Bai, Tian Ye, Wei Chow, Enxin Song, Qing-Guo Chen, Xiangtai Li, Zhen Dong, Lei Zhu, and Shuicheng Yan. Meissonic: Revitalizing masked generative transformers for efficient high-resolution text-to-image synthesis. In *ICLR*, 2025.

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL Technical Report, February 2025.

[4] Randall Balestriero, Nicolas Ballas, Mike Rabbat, and Yann LeCun. Gaussian Embeddings: How JEPAs Secretly Learn Your Data Density. *arXiv preprint arXiv:2510.05949*, 2025.

[5] Roy Betser, Eyal Gofer, Meir Yossef Levi, and Guy Gilboa. InfoNCE Induces Gaussian Distribution. In *ICLR*, 2026.

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9650–9660, 2021.

[7] Huiwen Chang, Han Zhang, Jarred Barber, Aaron Maschinot, José Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Patrick Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-To-Image Generation via Masked Generative Transformers. In *ICML*, volume 202, pages 4055–4075, 2023.

[8] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25(70):1–53, 2024.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[10] Boyang Ding, Chenglong Chu, Dunju Zang, Han Li, Jiangxia Cao, Kun Gai, Muhao Wei, Ruiming Tang, Shiyao Wang, Siyang Mao, Xinchen Luo, Yahui Liu, Zhixin Ling, Zhuoran Yang, Ziming Li, Chengru Song, Guorui Zhou, Guowang Zhang, Hao Peng, Hao Wang, Jiaxin Deng, Jin Ouyang, Jinghao Zhang, Lejian Ren, Qianqian Wang, Qigen Hu, Tao Wang, Xingmei Wang, Yiping Yang, Zixing Zhang, and Ziqi Wang. Kelix Technical Report. *arXiv preprint arXiv:2602.09843*, 2026.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[12] Sinan Du, Jiahao Guo, Bo Li, Shuhao Cui, Zhengzhuo Xu, Yifu Luo, Yongxian Wei, Kun Gai, Xinggang Wang, Kai Wu, et al. Vqrae: Representation quantization autoencoders for multimodal understanding, generation and reconstruction. *arXiv preprint arXiv:2511.23386*, 2025.

[13] Patrick Esser, Robin Rombach, and Björn Ommer. Taming Transformers for High-Resolution Image Synthesis. In *CVPR*, pages 12873–12883, 2021.

[14] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[15] Zigang Geng, Yibing Wang, Yeyao Ma, Chen Li, Yongming Rao, Shuyang Gu, Zhao Zhong, Qinglin Lu, Han Hu, Xiaosong Zhang, Linus, Di Wang, and Jie Jiang. X-Omni: Reinforcement Learning Makes Discrete Autoregressive Image Generative Models Great Again. *arXiv preprint arXiv 2507.22058*, 2025.

[16] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *NeurIPS*, 36:52132–52152, 2023.

[17] Jiaming Han, Hao Chen, Yang Zhao, Hanyu Wang, Qi Zhao, Ziyan Yang, Hao He, Xiangyu Yue, and Lu Jiang. Vision as a dialect: Unifying visual understanding and generation via text-aligned representations. *arXiv preprint arXiv:2506.18898*, 2025.

[18] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 15979–15988, 2022.

[19] 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. In *NeurIPS*, pages 6626–6637, 2017.

[20] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

[21] 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.

[22] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[23] Michel Ledoux. *The concentration of measure phenomenon*. Number 89. American Mathematical Soc., 2001.

[24] Meir Yossef Levi and Guy Gilboa. The double-ellipsoid geometry of CLIP. In *ICML*, volume 267, pages 33999–34019, 2025.

[25] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

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

[27] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

[28] Haokun Lin, Teng Wang, Yixiao Ge, Yuying Ge, Zhichao Lu, Ying Wei, Qingfu Zhang, Zhenan Sun, and Ying Shan. Toklip: Marry visual tokens to CLIP for multimodal comprehension and generation. *CoRR*, abs/2505.05422, 2025.

[29] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *ECCV*, pages 366–384. Springer, 2024.

[30] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *CVPR*, pages 26296–26306, 2024.

[31] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *ECCV*, pages 216–233. Springer, 2024.

[32] Chuofan Ma, Yi Jiang, Junfeng Wu, Jihan Yang, Xin Yu, Zehuan Yuan, Bingyue Peng, and Xiaojuan Qi. Unitok: A unified tokenizer for visual generation and understanding. *arXiv preprint arXiv:2502.20321*, 2025.

[33] Tianren Ma, Xiaosong Zhang, Boyu Yang, Junlan Feng, and Qixiang Ye. ReDDiT: Rehashing Noise for Discrete Visual Generation. In *ICLR*, 2026.

[34] Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *ICLR*, 2024.

[35] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024.

[36] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, pages 4172–4182, 2023.

[37] Wujian Peng, Lingchen Meng, Yuxuan Cai, Xianwei Zhuang, Yuhuan Yang, Rongyao Fang, Chenfei Wu, Junyang Lin, Zuxuan Wu, and Shuai Bai. Unified Multimodal Autoregressive Modeling with Shared Context-Visual Tokenizer is Key to Unification. *arXiv preprint arXiv:2606.18249*, 2026.

[38] 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. In *CVPR*, pages 2545–2555, 2025.

[39] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

[40] 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 *ICML*, pages 8748–8763, 2021.

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10674–10685, 2022.

[42] Subham S. Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and Effective Masked Diffusion Language Models. In *NeurIPS*, 2024.

[43] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, pages 2226–2234, 2017.

[44] Fengyuan Shi, Zhuoyan Luo, Yixiao Ge, Yujiu Yang, Ying Shan, and Limin Wang. Scalable Image Tokenization with Index Backpropagation Quantization. *arXiv preprint arXiv:2412.02692*, 2025.

[45] 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*, pages 8317–8326, 2019.

[46] Jaskirat Singh, Xingjian Leng, Zongze Wu, Liang Zheng, Richard Zhang, Eli Shechtman, and Saining Xie. What matters for representation alignment: Global information or spatial structure? In *ICLR*, 2026.

[47] Jaskirat Singh, Boyang Zheng, Zongze Wu, Richard Zhang, Eli Shechtman, and Saining Xie. Improved baselines with representation autoencoders. *arXiv preprint arXiv:2605.18324*, 2026.

[48] 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.

[49] Hao Tang, Chenwei Xie, Xiaoyi Bao, Tingyu Weng, Pandeng Li, Yun Zheng, and Liwei Wang. Unilip: Adapting clip for unified multimodal understanding, generation and editing. *arXiv preprint arXiv:2507.23278*, 2025.

[50] Meituan LongCat Team. LongCat-Next: Lexicalizing Modalities as Discrete Tokens. *arXiv preprint arXiv:2603.27538*, 2026.

[51] Shengbang Tong, David Fan, John Nguyen, Ellis Brown, Gaoyue Zhou, Shengyi Qian, Boyang Zheng, Th'eophane Vallaeys, Junlin Han, Rob Fergus, Naila Murray, Marjan Ghazvininejad, Mike Lewis, Nicolas Ballas, Amir Bar, Michael Rabbat, Jakob Verbeek, Luke S. Zettlemoyer, Koustuv Sinha, Yann LeCun, and Saining Xie. Beyond language modeling: An exploration of multimodal pretraining. In *ICML*, 2026.

[52] Shengbang Tong, Boyang Zheng, Ziteng Wang, Bingda Tang, Nanye Ma, Ellis Brown, Jihan Yang, Rob Fergus, Yann LeCun, and Saining Xie. Scaling text-to-image diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2601.16208*, 2026.

[53] Michael Tschannen, Alexey A. Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier J. Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *CoRR*, abs/2502.14786, 2025.

[54] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 30, 2017.

[55] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[56] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, volume 119, pages 9929–9939, 2020.

[57] Yuqing Wang, Zhijie Lin, Ceyuan Yang, Yang Zhao, Fei Xiao, Hao He, Qi Zhao, Zihan Ding, Fuyun Wang, Shuai Wang, Youliang Zhang, Haoqi Fan, and Xihui Liu. Representation Forcing for Bottleneck-Free Unified Multimodal Models. *arXiv preprint arXiv:2605.31604*, 2026.

[58] 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.

[59] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, Song Han, and Yao Lu. VILA-U: a unified foundation model integrating visual understanding and generation. In *ICLR*, 2025.

[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.

[61] Rongchang Xie, Chen Du, Ping Song, and Chang Liu. Muse-vl: Modeling unified vlm through semantic discrete encoding. In *ICCV*, pages 24135–24146, 2025.

[62] Jingfeng Yao, Yuda Song, Yucong Zhou, and Xinggang Wang. Towards scalable pre-training of visual tokenizers for generation. *arXiv preprint arXiv:2512.13687*, 2025.

[63] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *CVPR*, pages 15703–15712, 2025.

[64] Lijun Yu, José Lezama, Nitesh B. Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, Alexander G. Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A. Ross, and Lu Jiang. Language Model Beats Diffusion – Tokenizer is Key to Visual Generation. *arXiv preprint arXiv:2310.05737*, 2024.

[65] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. *arXiv preprint arXiv:1801.03924*, 2018.

[66] Yue Zhao, Hanwen Jiang, Zhenlin Xu, Chutong Yang, Ehsan Adeli, and Philipp Krähenbühl. Spherical leech quantization for visual tokenization and generation. *CoRR*, abs/2512.14697, 2025.

[67] 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.

[68] Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *CoRR*, abs/2510.11690, 2025.

[69] Yongxin Zhu, Bocheng Li, Yifei Xin, Zhihua Xia, and Linli Xu. Addressing representation collapse in vector quantized models with one linear layer. In *ICCV*, pages 22968–22977, 2025.

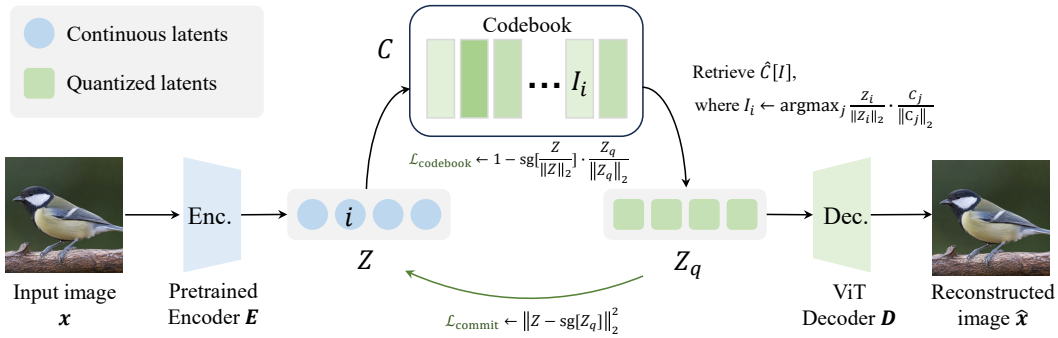

Figure 5: A demonstration of optimizing the codebook embeddings, following VQ-VAE [54].

A Implementation Details

A.1 Tokenizer Training

Algorithm The complete training algorithm for HSQ is demonstrated in Alg. 2. We also provide an illustration in Fig 5. The training is conducted on 8×80G NVIDIA GPUs for 3 days.

Algorithm 2 Hyper-Spherical Quantization

Input: Feature tensor $Z \in \mathbb{R}^{N \times d}$. Base codebook embeddings $\mathcal{C}$, learnable projection W, and commitment weight β
$\hat{C} \leftarrow W\mathcal{C}$ `# Project codebook following SimVQ`
$S_{i,j} \leftarrow \frac{Z_i}{\|Z_i\|_2} \cdot \frac{\hat{C}_j}{\|\hat{C}_j\|_2}$ `# Compute cosine similarity`
$I_i \leftarrow \arg\max_j S_{i,j}$ `# Angular routing`
$Z_q \leftarrow \hat{C}[I]$ `# Lookup on projected codebook`
$\mathcal{L}_{\text{codebook}} \leftarrow 1 - \text{sg}[\frac{Z}{\|Z\|_2}] \cdot \frac{Z_q}{\|Z_q\|_2}$ `# Angular codebook loss`
$\mathcal{L}_{\text{commit}} \leftarrow \|Z - \text{sg}[Z_q]\|_2^2$ `# Commitment loss`
$\mathcal{L}_{\text{VQ}} \leftarrow \mathcal{L}_{\text{codebook}} + \beta\mathcal{L}_{\text{commit}}$
$Z_{\text{out}} \leftarrow Z + \text{sg}[Z_q - Z]$ `# Straight-through estimator`
Return: $Z_{\text{out}}, \mathcal{L}_{\text{VQ}}, I$

Hyperparameter The training details of dRAE are provided in Tab. 12. The perceptual loss design follows the implementation of VQRAE, while the discriminator design follows RAE.

A.2 Additional Results

We set a variant of dRAE at 256×256 resolution, using DINOv2-B [35] as the encoder, a 12 layer ViT as the decoder, and train it for 20000 steps with 512 global batch size. The ablation results are shown in Tab. 10. We also follow RAEv2 [47] and test a variant that adds multiple layer features of the encoder for better reconstruction quality. Specifically, we use the sum of hidden states from DINOv2's `{3,6,9,12}` layers. The corresponding results are shown in Tab. 11. These experiments indicate that HSQ can also effectively capture the distribution of self-supervised visual encoders.

A.3 MLLM Training

Data usage and fair comparison For the main results table, the MLLM is trained using a combination of the publicly available datasets from LLaVA-1.5 and 1M randomly sampled image-text pairs from LLaVA-OneVision. We additionally train a controlled variant using a 384-resolution encoder, Vicuna-7B as the LLM backbone, and only the LLaVA-1.5 data. The comparison with other academic works is shown in Tab. 14. Under this more standardized training setup, our model—built upon an

Table 10: Ablation on reconstruction tasks with DINOv2-B Encoder.

Assign	$\mathcal{L}_{\text{codebook}}$	$\mathcal{L}_{\text{commit}}$	rFID	SSIM
L_2	L_2	L_2	3.67	0.56
θ	L_2	L_2	4.23	0.57
θ	θ	L_2	**2.87**	**0.58**

Table 11: Ablation on reconstruction tasks with RAEv2-style DINOv2-B Encoder.

Assign	$\mathcal{L}_{\text{codebook}}$	$\mathcal{L}_{\text{commit}}$	rFID	SSIM
L_2	L_2	L_2	4.24	0.56
θ	L_2	L_2	4.11	0.58
θ	θ	L_2	**3.29**	**0.60**

Table 12: Details for tokenizer training.

Category	Hyperparameter	Value
Architecture	Image Encoder	SigLIP2-So400M
	Encoder Input Size	512
	Decoder Type	ViT-XL
Training	Global Steps	150000
	Global Batchsize	512
	Optimizer	AdamW $(0.9, 0.95)$
	Learning Rate	2.0×10^{-4}
	Encoder Learning Rate	2.0×10^{-5}
	Weight Decay	0.0
	EMA Decay	0.9978
LR Scheduler	Type	Cosine
	Warmup Epochs	1
	Final Learning Rate	2.0×10^{-5}
Discriminator	Disc. Learning Rate	5.0×10^{-5}
	Disc. Loss Type	Hinge
	Gen. Loss Type	Vanilla
Loss Weights	Perceptual Weight ω_{p}	1.0
	Discriminator Weight ω_{d}	0.1
	Distillation Weight λ	1.0

Table 13: Details for generative models.

Category	Hyperparameter	Value
Architecture	Text Encoder	T5-XXL
	Max Text Length	64
	Image Resolution	384
	Decoder Type	DiT-H
	Codebook	65536
Training	Global Steps	125000
	Global Batchsize	512
	t Distribution	Logit Norm
	Optimizer	AdamW $(0.9, 0.95)$
	Learning Rate	2.0×10^{-4}
	Weight Decay	0.0
	EMA Decay	0.9999
LR Scheduler	Type	Constant
	Warmup Epochs	1
Inference	Steps	50
	Schedule	Cosine
	CFG	3.5

Table 14: **Evaluation on multimodal understanding**. All models use Vicuna-7B as the backbone and only LLaVA-1.5 data.

Method	Vision Encoder	Res.	POPE	GQA	TQA	MMB	SEED
LLaVA-1.5 [30]	CLIP ViT-L	336	85.9	62.0	46.1	64.3	58.6
VILA-U [59]	SigLIP ViT-So	384	85.8	60.8	60.8	-	59.0
UniTok [32]	Vitamin-L	256	81.7	-	-	-	-
QLIP [67]	CLIP ViT-L	392	86.1	61.8	55.2	-	-
TokLIP [28]	SigLIP2 ViT-So	384	84.1	59.5	-	67.6	70.4
dRAE	SigLIP2 ViT-So	384	85.5	61.0	60.1	71.7	65.0

encoder fine-tuned for reconstruction—still demonstrates advantages across multiple comprehension metrics, further validating the potential of dRAE as a unified tokenizer.

A.4 Generative Model Training

Discrete Diffusion Model (DDM) DDM defines a forward process on discrete variables by gradually corrupting tokens to absorbing state $\mathbf{m}$ through a continuous-time Markov process [42]. We denote clean data as $x_{t=0}$ (x_0 for short), and noise it gradually as $t \to 1$. Let α_t be the noise scheduler (a monotonically decreasing survival function that satisfies $\alpha_0 = 1, \alpha_1 = 0$), the corrupted data distribution at time t is determined as

$$x_t \sim q(x_t|x_0, t), q(x_t|x_0, t) = \text{Cat}(x_t; \alpha_t x_0 + (1 - \alpha_t)\mathbf{m}). \tag{7}$$

Let $\delta(x_{(t,i)}, \mathbf{m})$ be the indicator function that is activated only if the i-th position of x_t is $\mathbf{m}$. For a linear scheduler, the objective is derived as the evidence lower bound (ELBO) of $\log \pi_\theta(x_0|x_t)$:

$$\mathcal{L}_{\text{DDM}} = -\mathbb{E}_{t,\, x_0,\, x_t}[\frac{1}{t}\sum_{i=1}^{L} \delta(x_{(t,i)}, \mathbf{m}) \log \pi_\theta(x_{(0,i)}|x_t)] = -\mathbb{E}_{t,\, x_0,\, x_t}[\ell_{\pi_\theta}(x_t, x_0)]. \tag{8}$$

For conditional generation where a prompt $\mathbf{c}$ is given, we write $\ell_{\pi_\theta}(x_t, x_0|\mathbf{c})$ for simplicity. Following MDLM's deduction, assume that the network can reconstruct x_0 perfectly, we use $\pi_\theta(x_t)$ to

approximate this denoising process, and get the sampling rule as

$$p_\theta(x_s|x_t) = \begin{cases} 1, & \text{if } x_s = x_t,\ x_t \neq \mathbf{m}, \\ \frac{1-\alpha_s}{1-\alpha_t}, & \text{if } x_s = \mathbf{m},\ x_t = \mathbf{m}, \\ \frac{\alpha_s-\alpha_t}{1-\alpha_t}\pi_\theta(x_t), & \text{if } x_s \neq \mathbf{m},\ x_t = \mathbf{m}, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Hyperparameter The details of T2I generative training are provided in Tab. 13. Since a resolution of 512 requires 1024 tokens per image, the batch size becomes constrained on each machine. Therefore, we additionally train a dRAE with SigLIP2 ViT-large-patch16-384, which compresses each image into 576 tokens, enabling more efficient generative training. In addition, since a large vocabulary also significantly increases the memory footprint of the output layer, we set the tokenizer vocabulary size to 65,536. The training is conducted on 16 GPUs for 3 days. We provide qualitative results in Fig 6.

For C2I generation, we use SigLIP2 ViT-base-patch16-256 as the visual encoder, and train the tokenizer at 256 resolution, which means compressing each image into 256 tokens. The generative model is a 12-layer transformer.

A.5 Evaluation

Reconstruction We follow the evaluation protocol of RAE and assess reconstruction performance on the ImageNet validation set. During the reproduction of prior work, we observed that the SSIM evaluation in UniLIP contains a bug: the `data-range` passed to the evaluator is incorrectly set to 2.0 instead of the correct value of 1.0 corresponding to the actual image dynamic range, leading to an overestimation of SSIM by $\sim 10\%$. The VQRAE implementation, based on the same codebase, suffers from the same issue. Accordingly, we correct the reported SSIM values for both methods based on our experiments.

Understanding We use the official script in the repository of LLaVA-1.5 for evaluation.

Generation We follow the standard protocol of GenEval and DPG-Bench, which samples 4 images for each prompt and use their evaluation model to score. The images are sampled with 50 steps using a cosine schedule and a constant classifier free guidance (CFG) of 3.5.

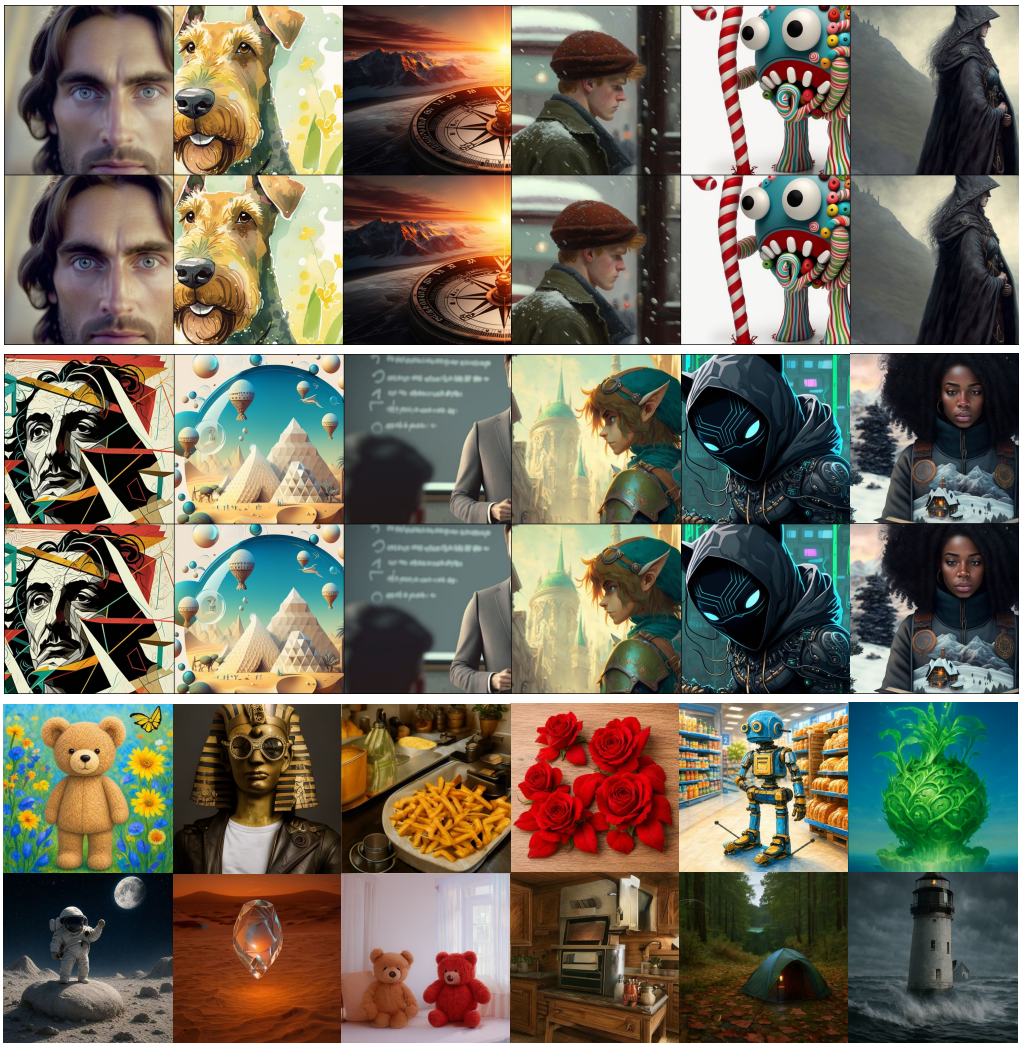

Figure 6: **Qualitative results for reconstruction and generation.** From the first four rows, the input images are shown above and the corresponding reconstructed images are shown below. The last two rows present generated samples.

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 clearly state the claims made.

   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: We discussed the experiment limitation explicitly.

   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: We provided necessary citation to our motivation origins.

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: The appendix has included algorithm, training hyperparameter and evaluation details.

   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: [Yes]

Justification: All of the data used is publicly available. The code is also based on open-sourced repositories.

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: Details are in the Appendix.

   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: [N/A]

   Justification: We follow common practice on evaluation metrics, where the statistical significance is omitted.

   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: Details are in the Appendix.

   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: The research conducted does conform 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: [N/A]

    Justification: As a quantization method research, there is no explicit societal impact. The downstream tasks may have, but is not the focus of our research.

    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: [N/A]

    Justification: The paper poses no such risks.

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

    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: [Yes]

Justification: We use LLM for simple polishing, *not* for impact the core methodology, scientific rigor, or originality of the research. And the usage is discussed explicitly in the appendix.

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.