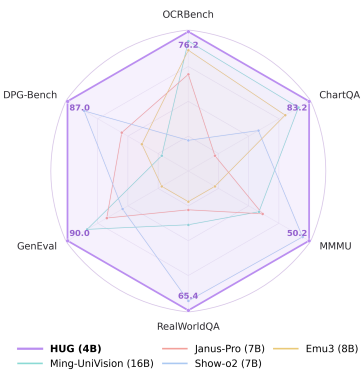
HUG: LEARNING AND TRANSFERRING VISUAL TOKENS FOR UNIFIED IMAGE UNDERSTANDING AND GENERATION

Anonymous authors
Paper under double-blind review



(a) Comparison with unified models.



(b) Image generation examples.

Figure 1: **HUG understanding and generation results.** (a) Benchmark comparison with unified multimodal models. (b) Images generated by HUG.

ABSTRACT

We present **HUG**, a framework for unified image understanding and autoregressive image generation that combines visual tokenizer learning with vocabulary adaptation to a pretrained LLM. A shared vision encoder provides continuous features for understanding and discrete tokens for generation. HUG supplements post-quantization caption supervision and discrete-to-continuous alignment with direct caption and masked VAE-latent prediction before quantization. Controlled experiments show that direct pre-quantization supervision improves reconstruction fidelity and code-use diversity. Increasing the post-caption loss weight does not reproduce the pre-caption gains, and pre-quantization masked prediction outperforms its post-quantization counterpart in our setting. For vocabulary transfer, HUG builds on text-scale-aware normalization and uses Text-Aligned Normalization (TAN) with caption-supervised adaptation to initialize the LLM's visual input and output weights. With the tokenizer fixed, the full TAN procedure outperforms the tested fixed and learned initialization alternatives without adding a downstream module. With a 0.6B Qwen3 language backbone, HUG achieves 0.85 on GenEval and 86.05 on DPG-Bench. Scaling to 4B yields 0.90 and 87.00, respectively, the highest overall scores on both benchmarks among the unified models evaluated, alongside competitive multimodal understanding.

1 INTRODUCTION

Multimodal large language models have made substantial progress in visual understanding (Team et al., 2023; Bai et al., 2025; Wen et al., 2026; Team et al., 2025), alongside rapid advances in image generation (Rombach et al., 2022; Seedream et al., 2025; Wu et al., 2025a). Recent work seeks to combine these capabilities within a single model (Chen et al., 2025d; Deng et al., 2025a; Li et al., 2026b; Liu et al., 2025). Discrete visual tokens provide one approach: a language backbone predicts textual responses and image-token sequences autoregressively, while an image decoder renders the predicted tokens into pixels (Team, 2024; Cui et al., 2025; Zigang Geng & Jiang, 2025).

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

The resulting system depends on both the information preserved by its visual tokenizer and how the learned vocabulary is introduced into the pretrained language model.

Considerable effort has been devoted to learning better discrete visual representations. Existing approaches explore larger or multiple codebooks (Ma et al., 2026a; Zhuang et al., 2026), caption-supervised tokenization (Zigang Geng & Jiang, 2025; Li et al., 2026b), and feature-reconstruction objectives (Han et al., 2025a). In several closely related methods, these objectives are applied to quantized features. Although their gradients also train the upstream network, the latent before quantization does not receive a separate prediction objective. This raises a concrete question: can directly supervising that latent improve visual information preservation beyond the existing post-quantization objectives? Caption prediction and visual reconstruction impose different requirements on a representation, and their placement relative to quantization is a design choice that merits explicit evaluation.

A second question arises after the tokenizer has been learned. Extending a pretrained LLM with visual token IDs requires input embeddings and output prediction weights for the new tokens. Reusing tokenizer embeddings offers a way to transfer learned token-specific representations, but these vectors were not directly optimized as the target LLM's input and output weights. Prior work studies how visual–text norm discrepancies affect representation updates in pre-norm MLLMs (Li et al., 2026a). In visual-token generation, transferred embeddings also serve as output classifier weights, whose scale affects prediction logits. These considerations motivate examining vocabulary initialization separately from tokenizer quality: a representation useful for encoding images need not be equally suitable for direct transfer into a pretrained LLM.

We present **HUG**, a framework that connects visual tokenizer learning with vocabulary adaptation for unified image understanding and autoregressive image generation. HUG uses a shared SigLIP2 encoder with a continuous branch for understanding and a discrete branch for generation. The discrete branch is trained with caption supervision and discrete-to-continuous (D2C) alignment after quantization. D2C uses unquantized features from the continuous branch as a visual reference, providing a feature-level constraint in addition to caption prediction. Rather than relying solely on these post-quantization objectives, HUG directly supervises the compact latent before finite scalar quantization (FSQ). A pre-quantization caption objective supplies a direct language-prediction signal, while masked VAE-latent prediction provides complementary visual supervision. To transfer the resulting vocabulary to a pretrained LLM, we build on text-scale-aware normalization (Li et al., 2026a) and use Text-Aligned Normalization (TAN) with caption-supervised adaptation. Only the transformed vocabulary is transferred to initialize the target LLM's visual input and output weights; the adapted auxiliary LLM weights are not retained. TAN is used only for initialization and adds no module during downstream training or inference.

Controlled ablations show that direct pre-quantization supervision improves reconstruction fidelity and code-use diversity, while TAN improves generation with the tokenizer fixed. With a 0.6B Qwen3 language backbone, HUG achieves 0.85 on GenEval (Ghosh et al., 2023) and 86.05 on DPG-Bench (Hu et al., 2024). Scaling the backbone to 4B raises these scores to 0.90 and 87.00, surpassing strong baselines such as UniWeTok-8B and BAGEL-14B on both benchmarks and achieving the highest overall scores among the unified models in our comparison. HUG also achieves competitive multimodal understanding within this group.

Our contributions are threefold:

- We develop a shared-encoder dual-branch tokenizer that augments post-quantization objectives with direct pre-quantization caption and masked VAE-latent supervision. Loss-reweighting and MIM-placement controls support this design through reconstruction and code-use analyses.
- We adapt text-scale-aware normalization to discrete visual-vocabulary transfer. TAN combines caption-supervised adaptation with initialization of the LLM's visual input and output weights, and we evaluate the roles of its initial parameterization and subsequent adaptation.
- We demonstrate strong generation and competitive understanding performance with 0.6B and 4B language backbones, supported by controlled studies of tokenizer supervision, vocabulary initialization, and component contributions under joint training.

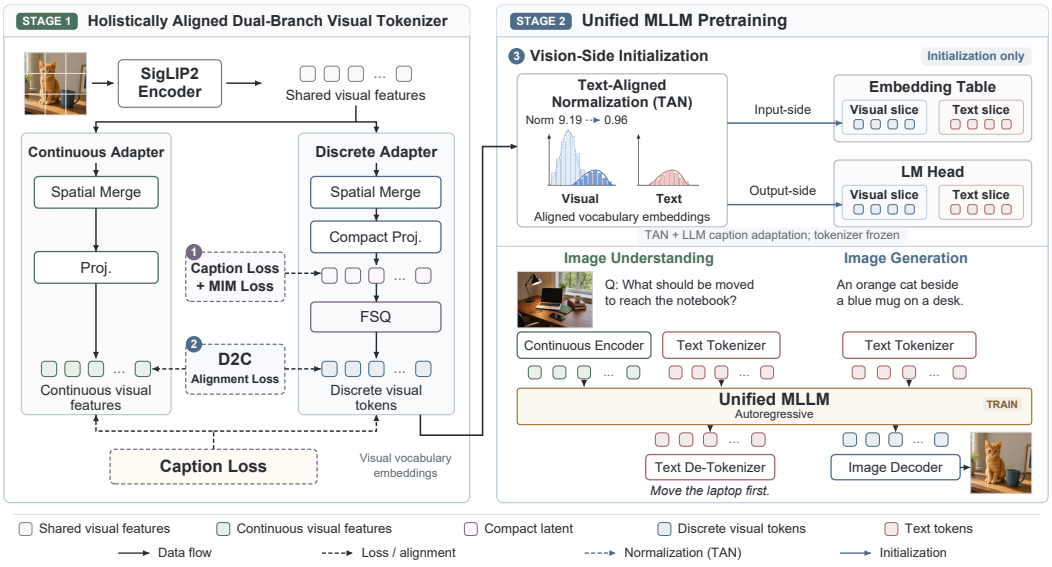


Figure 2: Overview of HUG for unified image understanding and autoregressive image generation. Stage 1 learns a dual-branch visual tokenizer on top of a shared SigLIP2 encoder, with continuous features for understanding and discrete tokens for generation. Stage 2 first performs caption-supervised TAN adaptation to initialize the LLM's visual input and output weights, then jointly trains the LLM on understanding and generation.

2 Related Work

Unified Multimodal Understanding and Generation. Unified multimodal models explore different modeling objectives and visual interfaces for understanding and generation. Chameleon (Team, 2024) and Emu3 (Wang et al., 2026c) model text and discrete image tokens through next-token prediction. Transfusion (Zhou et al., 2025), Show-o2 (Xie et al., 2026), TUNA (Liu et al., 2025) and TUNA2 (Liu et al., 2026) combine language autoregression with diffusion or flow-based visual generation. Visual-pathway design is a separate consideration: Janus (Wu et al., 2025b) and Janus-Pro (Chen et al., 2025d) separate the visual encoders for understanding and generation, while BAGEL (Deng et al., 2025b) and SenseNova-U1 (Diao et al., 2026) use task-specific experts. MANZANO (Li et al., 2026b) shares a vision encoder across continuous understanding and discrete generation adapters. HUG adopts a related dual-branch interface, focusing on tokenizer supervision and the transfer of its discrete vocabulary to the language backbone.

Discrete Visual Tokenizers. Recent work improves visual tokenization through code capacity and training supervision. UniTok (Ma et al., 2026a) uses multi-codebook quantization, while UniWeTok (Zhuang et al., 2026) and BitDance (Ai et al., 2026) explore large binary code spaces. X-Omni (Zigang Geng & Jiang, 2025) trains an IBQ-based tokenizer through caption supervision on quantized features; GLM-Image (Z.ai, 2026) builds on this tokenizer in an autoregressive-to-diffusion generation pipeline. Tar (Han et al., 2025a) applies feature-reconstruction supervision to quantized representations, and its tokenizer is subsequently used by BLIP3o-NEXT (Chen et al., 2025b). MANZANO (Li et al., 2026b) describes caption supervision on FSQ features, while UniAR (Peng et al., 2026) uses a shared BSQ-based tokenizer for understanding and generation. ARM (Wang et al., 2026a) combines multiple tokenizer objectives with autoregressive training and reinforcement learning for generation and editing. HUG additionally places caption and masked VAE-latent prediction objectives directly on the compact latent before quantization, while retaining post-quantization caption supervision and D2C alignment. Our loss-reweighting and MIM-placement controls examine the value of this direct pre-quantization supervision, beyond learning upstream representations through post-quantization objectives alone.

Visual–Language Interface Adaptation. Tokenizer learning and vocabulary transfer address different aspects of integrating visual tokens into a pretrained LLM. Prior work (Li et al., 2026a) analyzes visual–text norm discrepancies in pre-norm MLLMs and introduces text-scale-aware normalization to improve visual-input processing. TAN builds on this principle for discrete visual-vocabulary transfer.

We adapt a vocabulary transformation through caption supervision and use the resulting embeddings to initialize the LLM's visual input and output weights. The adapted auxiliary LLM weights are not transferred, and TAN is not retained as a downstream module. With the tokenizer fixed, our experiments compare fixed normalization, text-scale-aware initialization without adaptation, and caption-adapted alternatives.

3 METHOD

HUG follows a two-stage framework, as illustrated in Figure 2. In Stage 1, we learn a dual-branch visual tokenizer on top of a shared SigLIP2 encoder. The continuous branch produces visual features for image understanding, while the discrete branch maps an image to autoregressively predictable visual tokens for image generation. The discrete branch is supervised on both sides of FSQ: caption prediction and masked latent-to-VAE prediction shape the compact latent before quantization, while post-quantization caption prediction and discrete-to-continuous alignment regularize the resulting quantized features.

In Stage 2, we first adapt the learned discrete visual vocabulary to the pretrained language model with Text-Aligned Normalization (TAN). The adapted vocabulary initializes the visual input embedding slice and the visual-token output slice of the target LLM. We then perform unified multimodal training, using continuous visual features to generate textual responses for image understanding and discrete visual tokens to autoregressively generate images from text prompts. The predicted visual tokens are finally rendered into images by the image decoder.

3.1 DUAL-BRANCH VISUAL TOKENIZER

Given an input image I, the shared SigLIP2 encoder extracts patch-level visual features

$$\mathbf{h} = f_{\text{SigLIP2}}(I) \in \mathbb{R}^{N \times d_v}, \tag{1}$$

where N is the number of visual patches. We apply two task-specific adapters to the shared features: a continuous adapter for image understanding and a discrete adapter for autoregressive image generation. The training objectives of the discrete adapter before and after FSQ are summarized in Figure 3.

Shared features and visual representations. The continuous adapter applies limited spatial compression to $\mathbf{h}$ and projects the resulting features to the hidden dimension of the auxiliary LLM. We denote its output by

$$\mathbf{c} = f_{\text{cont}}(\mathbf{h}) \in \mathbb{R}^{M \times D_c}, \tag{2}$$

where M is the number of visual token positions and D_c is the feature dimension used for the continuous understanding pathway. The continuous features $\mathbf{c}$ are provided to the LLM for image understanding and also serve as the unquantized visual reference in the D2C objective.

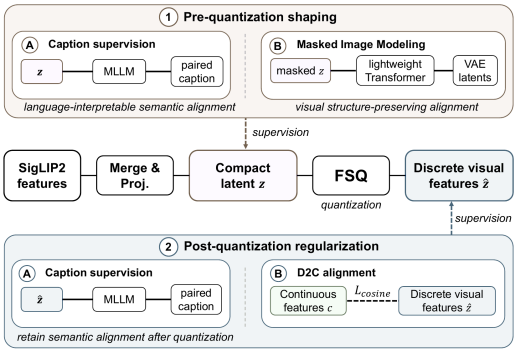


Figure 3: Training of the dual-branch visual tokenizer before TAN adaptation. HUG applies caption and masked latent-to-VAE supervision before FSQ, and caption and D2C supervision after FSQ.

The discrete adapter first performs spatial merging and a compact projection to obtain the pre-quantization latent

$$\mathbf{z} = f_{\text{disc}}(\mathbf{h}) \in \mathbb{R}^{M \times 512}. \tag{3}$$

The latent $\mathbf{z}$ is then quantized by FSQ. At each spatial position, FSQ produces a discrete code index together with its corresponding quantized feature. We denote the resulting token indices by $\mathbf{s}$. The quantized features are subsequently projected to

$$\mathbf{d} = f_{\text{q}}(\mathbf{z}) \in \mathbb{R}^{M \times D_c}. \tag{4}$$

Thus, $\mathbf{s}$ is a sequence of discrete token IDs, whereas $\mathbf{d}$ is the continuous feature associated with the quantized representation. The token indices $\mathbf{s}$ are used as autoregressive prediction targets, while $\mathbf{d}$ is used for post-quantization supervision, D2C alignment, and visual-vocabulary construction.

The continuous and quantized features have the same dimensionality and are compared at corresponding spatial positions. The quantization and image-decoder architectures are the same across tokenizer ablations. For each tokenizer variant, we separately train a decoder using the same training configuration. The tokenizer variants differ in the supervision objectives applied before and after FSQ.

Post-quantization supervision. As illustrated in Figure 3, we first supervise the projected quantized features $\mathbf{d}$ after FSQ. A dedicated projection MLP maps $\mathbf{d}$ to the hidden dimension of the shared auxiliary LLM, which predicts the paired image caption. This objective encourages the quantized features to retain information useful for caption prediction after discretization.

Caption prediction alone does not explicitly constrain the quantized features to preserve information represented by the corresponding unquantized visual features. We therefore use the continuous branch as a visual reference for the quantized branch and define the discrete-to-continuous alignment loss

$$\mathcal{L}_{\mathrm{D2C}} = 1 - \frac{1}{M}\sum_{i=1}^{M} \cos(\mathbf{d}_i, \mathrm{sg}(\mathbf{c}_i)), \tag{5}$$

where $\mathbf{d}_i$ and $\mathbf{c}_i$ denote the projected quantized and continuous features at the corresponding spatial positions, and $\mathrm{sg}(\cdot)$ denotes stop-gradient. The continuous features provide the alignment targets, while the D2C loss updates the quantized branch without propagating gradients through the target features. Thus, D2C provides a feature-level visual constraint in addition to caption supervision and encourages the quantized representation to retain information from the unquantized visual reference.

Direct pre-quantization supervision. The post-quantization objectives supervise the representation after FSQ and can update upstream parameters through the quantization path. However, this does not make direct supervision of the pre-quantization latent redundant. In our experiments, post-quantization caption and D2C supervision alone produces a relatively concentrated code-use distribution. Adding caption supervision directly to $\mathbf{z}$ improves both code-use perplexity and reconstruction (Table 3(a)). Motivated by this observation, we directly supervise $\mathbf{z}$ before FSQ with two complementary prediction objectives.

First, we apply the same auxiliary captioning task before quantization. A separate MLP maps $\mathbf{z}$ to the hidden dimension of the shared auxiliary LLM, which predicts the paired image caption. This provides a direct caption-prediction signal to the representation entering FSQ, while the post-quantization caption objective continues to supervise the resulting quantized features.

Second, we introduce masked latent-to-VAE prediction, which we refer to as pre-MIM. We randomly mask a subset of positions in $\mathbf{z}$ and feed the corrupted sequence to an auxiliary Transformer. The Transformer predicts the corresponding VAE latents, and the mean squared error is computed over all target positions, including both masked and unmasked locations:

$$\mathcal{L}_{\text{pre-MIM}} = \frac{1}{M}\sum_{i=1}^{M} \|\widehat{\mathbf{v}}_i - \mathbf{v}_i\|_2^2, \tag{6}$$

where $\mathbf{v}_i$ denotes the target VAE latent and $\widehat{\mathbf{v}}_i$ denotes the prediction from the masked sequence. The VAE latent provides a compact visual prediction target, allowing this auxiliary objective to supervise local appearance and structure without introducing a full-resolution image reconstruction decoder.

Tokenizer optimization. The three visual pathways share an auxiliary LLM. At each step, we sample a pathway r and compute a single caption loss $\mathcal{L}_{\mathrm{cap}}^{(r)}$. The training objective is

$$\mathcal{L} = w_r \mathcal{L}_{\mathrm{cap}}^{(r)} + \lambda_{\mathrm{MIM}}\mathcal{L}_{\mathrm{MIM}} + \lambda_{\mathrm{D2C}}\mathcal{L}_{\mathrm{D2C}}, \tag{7}$$

where $w_r = 1$, $\lambda_{\mathrm{MIM}} = 0.2$, and $\lambda_{\mathrm{D2C}} = 1$ in the full tokenizer. Disabled auxiliary objectives are omitted in the corresponding ablations. The pathway-sampling schedules and post-caption reweighting control are detailed in Appendix A.1.

For the 4B backbone, we reuse the tokenizer learned with the 0.6B auxiliary LLM, but replace the continuous-branch and post-quantization projectors with newly initialized projectors matching the 4B hidden dimension. We train these two projectors using caption supervision on their respective visual features, while keeping the remaining tokenizer components frozen. Pre-quantization caption prediction, MIM, and D2C are not used during this projector-adaptation stage.

3.2 TEXT-ALIGNED VISUAL VOCABULARY INITIALIZATION

To enable autoregressive prediction of discrete visual tokens, we extend the pretrained LLM's vocabulary with the token IDs defined by the visual tokenizer. Each added token requires an input embedding and a corresponding output prediction weight. Rather than assigning these weights random values, we use the tokenizer's learned vocabulary embeddings as the starting point, reusing the representations associated with each visual token.

Direct transfer, however, need not yield weights suited to the target LLM. For Qwen3-0.6B, the visual vocabulary embeddings before adaptation have an average ℓ_2 norm of 9.19, compared with 0.93 for the pretrained text input embeddings. Prior work studies how visual–text norm discrepancies affect visual-input processing in pre-norm MLLMs (Li et al., 2026a). At the output interface, the transferred vectors serve as classifier weights, whose scale also affects the logits used for visual-token prediction. Our ablations show that L2-normalized initialization of the tied visual input and output weights improves generation over raw vocabulary transfer (Table 3(b)). This motivates adapting the learned vocabulary before using it to initialize the LLM's visual input and output weights. Building on the text-scale-aware normalization approach of Li et al. (2026a), we use Text-Aligned Normalization (TAN) to adapt the vocabulary before initializing the LLM's visual input and output weights.

Text-scale-aware parameterization. Let $\mathbf{W}_e$ denote the target LLM's original pretrained text embedding matrix. We estimate its reference scale from non-zero rows:

$$T = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} \|\mathbf{W}_e[j]\|_2, \qquad \mathcal{J} = \{j : \|\mathbf{W}_e[j]\|_2 > \epsilon\}. \tag{8}$$

For a visual vocabulary embedding $\mathbf{x} \in \mathbb{R}^D$ projected to the target LLM's hidden dimension, TAN applies

$$\mathrm{TAN}(\mathbf{x}) = \mathbf{g} \odot \frac{\mathbf{x} - \mu(\mathbf{x})}{\sqrt{\mathrm{Var}(\mathbf{x}) + \epsilon_{\mathrm{ln}}}} + \boldsymbol{\beta}, \qquad \mathbf{g}_0 = \frac{T}{\sqrt{D}}, \quad \boldsymbol{\beta}_0 = \mathbf{0}, \tag{9}$$

where the mean and variance are computed across embedding dimensions, and $\mathbf{g}, \boldsymbol{\beta} \in \mathbb{R}^D$ are learnable affine parameters. A standardized D-dimensional vector has norm approximately $\sqrt{D}$, so this gain initialization places the transformed embeddings near the native text scale. It sets the initial scale without constraining embedding norms during adaptation.

Caption adaptation and vocabulary transfer. With the tokenizer frozen, we optimize TAN together with an auxiliary LLM using image-caption loss: the LLM predicts the paired caption conditioned on the transformed visual embeddings. After adaptation, we retain only the transformed vocabulary and load the target LLM from its original pretrained checkpoint. The adapted auxiliary LLM weights are not transferred.

For both the 0.6B and 4B backbones, we tie the input embedding matrix and the output LM-head weights. Each added visual token therefore uses the same parameter vector for input lookup and output prediction. We initialize these shared visual weights from the tokenizer's learned vocabulary after TAN adaptation. Let $\mathbf{W}$ denote the tied input embedding and output weight matrix, and let $\mathcal{I}_v$ index the visual tokens in the expanded vocabulary. We initialize

$$\mathbf{W}[\mathcal{I}_v] \leftarrow \mathrm{TAN}(\mathbf{E}_{\mathrm{vision}}). \tag{10}$$

The transformed embeddings thus serve both as visual input representations and as output prediction weights. Weight tying is retained during unified training. TAN is used only to construct the initial weights and is not retained as an additional downstream module.

4 EXPERIMENTS

In this section, we first describe the training recipe of HUG in Section 4.1. We then present the main results on multimodal understanding and image generation benchmarks in Section 4.2. Finally, we provide ablation studies on tokenizer objectives and visual-vocabulary initialization in Section 4.3. Please refer to Appendix A for more training details and evaluation setup.

4.1 TRAINING RECIPE

HUG uses SigLIP2 as the visual encoder, Qwen3-0.6B or Qwen3-4B-Instruct as the language backbone, and Lumina-Image 2.0 (Qin et al., 2025) as the image decoder. We first train the dual-

Table 1: Evaluation on multimodal understanding benchmarks. Bold numbers indicate the best performance among the understanding-and-generation models listed in the table.

Model	Size	MMStar	MMBench-EN (Dev)	OCRBench	ChartQA	AI2D	MMMU	RealWorldQA
Understanding-Only Models								
LLaVA-OV-1.5 (An et al., 2025)	8B	67.7	82.6	82.9	86.5	84.2	55.4	54.8
Qwen3-VL (Bai et al., 2025)	8B	69.8	85.1	88.1	–	84.1	67.4	62.9
MiniCPM-V4.5 (Yao et al., 2024)	8B	–	87.4	87.6	–	87.6	67.6	–
Innovator-VL (Wen et al., 2026)	8B	65.0	85.3	-	–	85.6	55.2	71.5
Understanding & Generation								
Harmon (Wu et al., 2025c)	1.5B	35.3	65.5	11.2	29.8	57.0	38.9	49.8
Cheers (Zhang et al., 2026)	1.5B	50.9	70.4	58.4	75.7	74.4	36.0	60.9
Show-o2 (Xie et al., 2026)	7B	56.6	–	32.4	52.3	78.6	48.9	64.7
VILA-U (Wu et al., 2024)	7B	–	–	23.3	11.4	49.0	31.2	46.8
Janus-Pro (Chen et al., 2025d)	7B	48.3	–	59.0	25.8	71.3	41.0	58.0
Emu3 (Wang et al., 2026b)	8B	–	58.5	68.7	68.6	70.0	31.6	57.4
Ming-Univision (Huang et al., 2025)	16B	–	–	72.4	76.7	**82.8**	40.3	59.1
HUG	4B	**57.6**	**73.5**	**76.2**	**83.2**	79.8	**50.2**	**65.4**

branch tokenizer and its decoder, then perform caption-supervised TAN adaptation. The transformed vocabulary initializes a target LLM loaded from its original pretrained checkpoint, followed by unified pretraining, instruction tuning, and SFT. The decoder is conditioned only on visual-token features, without the text prompt. The full recipe combines public datasets with internal and synthetic data. Stage-specific datasets, training budgets, task sampling, and backbone-specific adaptation are detailed in Appendices A.2 and A.3.

4.2 Main Results

As shown in Table 1, HUG achieves competitive multimodal understanding performance among unified models across perception, text-rich understanding, and visual reasoning benchmarks. Meanwhile, Table 2 shows strong image generation performance at both backbone scales. With a 0.6B language backbone, HUG achieves 0.85 on GenEval and 86.05 on DPG-Bench. Scaling the backbone to 4B further improves these scores to 0.90 and 87.00, respectively, achieving the highest overall scores among the unified models compared in the table. Together, these results demonstrate HUG's ability to combine strong image generation with competitive multimodal understanding in a single model. We examine the contributions of the tokenizer objectives and vocabulary initialization through controlled ablations in Section 4.3.

4.3 Ablations

We examine two design choices: supervision around quantization and vocabulary initialization in the LLM. A third study evaluates component contributions under joint understanding-and-generation training. All downstream ablations use Qwen3-0.6B. Panel (b) uses generation-only training, whereas panel (c) jointly trains understanding and generation. Comparisons are controlled within each study; detailed protocols are provided in Appendix A.4. In panel (c), C.L. denotes both pre- and post-quantization caption supervision.

Supervision around quantization. We keep the image encoding and quantization architecture, training steps, and computational budgets fixed. For each tokenizer, we separately train a decoder with the same architecture and training configuration, and evaluate checkpoints at the same training step under a common reconstruction protocol. Reconstruction uses encoded image tokens, not LLM predictions; the evaluation protocol is detailed in Appendix A.4. Code-use perplexity measures the effective diversity of codes obtained from 200K sampled training images.

In Table 3(a), adding D2C to post-caption supervision improves PSNR/SSIM from 13.32/0.33 to 15.06/0.37. Doubling the post-caption weight yields similar reconstruction (15.02/0.36) and code-use perplexity (approximately 16K). In contrast, adding pre-caption with the original post-caption weight improves reconstruction to 15.73/0.42 and perplexity to approximately 25K. Thus, this reweighting control does not reproduce the gains from directly supervising the pre-quantization latent.

We further compare MIM placement while retaining pre-caption, post-caption, and D2C. Post-MIM reaches 16.50/0.56 and approximately 29K PPL, whereas pre-MIM reaches 18.25/0.60 and approximately 38K. Both placements improve reconstruction over the no-MIM variant, with larger gains before quantization. Together, these controls support direct pre-quantization supervision in our setting, with increased code-use diversity accompanied by improved reconstruction fidelity.

Table 2: Comparison of different models on GenEval (Ghosh et al., 2023) and DPG-Bench (Hu et al., 2024). Selected subcategory scores and overall scores are reported. Overall scores follow the full benchmark evaluation and are not recomputed from the displayed subcategories. Best results within the small-scale and large-scale UMM groups are highlighted in bold.

Models	Size	GenEval					DPG-Bench			
		Two.	Count.	Col.	Pos.	Overall	Global	Entity	Attr.	Overall
Generation-only Models										
SD3-Medium (Esser et al., 2024)	2B	0.94	0.72	0.89	0.33	0.74	87.90	91.01	88.83	84.08
FLUX.1[dev] (Labs et al., 2025)	12B	0.93	0.75	0.93	0.68	0.82	82.10	89.50	88.70	84.00
Qwen-Image (Wu et al., 2025a)	27B	0.92	0.89	0.88	0.76	0.87	91.32	91.56	92.02	88.32
Seedream3.0 (Gao et al., 2025)	–	0.96	0.91	0.93	0.47	0.84	94.31	92.65	91.36	88.27
GPT-4o	–	0.92	0.85	0.92	0.75	0.84	82.27	91.27	87.67	86.23
Small-scale UMMs										
Janus-Pro (Chen et al., 2025d)	1.5B	0.82	0.51	0.89	0.65	0.73	87.58	88.63	88.17	82.63
Show-o2 (Xie et al., 2026)	1.5B	0.86	0.55	0.86	0.46	0.73	87.53	90.38	**91.34**	85.02
Tar (Han et al., 2025a)	1.5B	0.91	0.76	0.81	0.57	0.76	83.59	89.35	86.91	82.96
Harmon (Wu et al., 2025c)	1.5B	0.86	0.66	0.85	0.74	0.76	–	–	–	–
HUG	0.6B	**0.94**	**0.81**	**0.91**	**0.77**	**0.85**	**92.55**	**92.51**	89.70	**86.05**
Large-scale UMMs										
BAGEL (Deng et al., 2025a)	14B	0.95	0.84	**0.95**	0.78	0.88	88.94	90.37	91.29	85.07
NextStep-1 (Han et al., 2025b)	14B	–	–	–	–	0.73	–	–	–	85.28
Ming-UniVision (Huang et al., 2025)	16B	0.93	0.59	0.93	**0.92**	0.85	–	–	–	82.12
Manzano (Li et al., 2026b)	30B	0.91	0.84	0.87	0.84	0.85	–	–	–	–
MetaQuery (Pan et al., 2025)	7B	–	–	–	–	0.80	82.10	79.30	77.20	77.40
Emu3 (Wang et al., 2026b)	8B	0.81	0.42	0.80	0.49	0.66	80.60	85.21	86.68	83.15
Janus-Pro (Chen et al., 2025d)	7B	0.89	0.59	0.90	0.79	0.80	86.90	88.90	89.40	84.19
BLIP-3o (Chen et al., 2025a)	8B	–	–	–	–	0.84	–	–	–	81.60
Show-o2 (Xie et al., 2026)	7B	0.87	0.58	0.92	0.52	0.76	89.00	91.78	89.96	86.14
Tar (Han et al., 2025a)	7B	0.92	0.83	0.85	0.80	0.84	–	88.62	88.05	84.19
UniWeTok (Zhuang et al., 2026)	8B	0.94	0.43	0.90	0.78	0.81	87.54	**91.99**	90.72	86.63
HUG	4B	**0.96**	**0.86**	0.94	0.86	**0.90**	**93.63**	91.87	**91.88**	**87.00**

Visual-vocabulary initialization. We fix the full tokenizer and decoder and initialize fresh copies of the same pretrained Qwen3-0.6B checkpoint for identical downstream T2I training. Caption-adapted variants first optimize their visual transformation and an auxiliary LLM using the same caption data, training budget, optimization settings, and auxiliary LLM update policy. Only the transformed vocabulary is transferred; the adapted auxiliary LLM weights are not retained.

Table 3(b) reports GenEval scores of 0.15, 0.19, and 0.35 for random, raw, and two-sided L2 initialization, respectively. Raw/L2 achieves 0.27 using a separate raw visual input table and L2-normalized output weights. Because this configuration is untied, its comparison with tied configurations does not isolate normalization from parameter sharing. All normalization operations construct the initial weights and are not enforced during downstream training.

The remaining controls examine fixed scale matching, gain initialization, and learned adaptation. Text-L2 rescales each unit-normalized embedding to the mean text embedding norm T and achieves 0.34, close to L2 at 0.35. TAN without caption adaptation also achieves 0.34, whereas the full procedure reaches 0.40. Thus, fixed text-scale normalization does not recover the full result, and caption adaptation provides an additional gain within the TAN parameterization.

Under matched adaptation settings, standard LayerNorm differs from TAN only in its initial gain (1 versus $T/\sqrt{D}$, with zero bias in both). It yields an average visual norm of approximately 28 and 0.13 GenEval, compared with 0.96 and 0.40 for TAN. This supports text-scale-aware initialization (Li et al., 2026a) as a starting point for vocabulary adaptation. Proj.+L2 adapts only the projector used to construct the transferred vocabulary on the visual side, with unit-normalized outputs, and achieves 0.36 under the same adaptation budget. The full TAN procedure therefore outperforms the tested fixed and learned alternatives.

Supplementary training curves for Raw, L2, and TAN (Figure 4, Appendix A.6) also show lower training loss with TAN toward the end of the evaluated budget, under settings matched across those runs. These results demonstrate that vocabulary initialization affects generation and optimization even when the tokenizer is unchanged.

Table 3: Ablations of HUG. (a) Tokenizer objectives, caption-loss reweighting, and MIM placement. (b) Vocabulary initialization with the full tokenizer from the last row of (a) fixed, followed by generation-only training (LR: 5×10^{-5}). (c) Joint understanding-and-generation training (LR: 1.5×10^{-4}). MMB-EN denotes MMBench-EN (Dev).

(a) Visual tokenizer quality

Post-cap weight	D2C	Pre-caption	MIM position	Code-use PPL	PSNR ↑	SSIM ↑
1	–	–	–	–	13.32	0.33
1	✓	–	–	≈ 16K	15.06	0.37
2	✓	–	–	≈ 16K	15.02	0.36
1	✓	✓	–	≈ 25K	15.73	0.42
1	✓	✓	Post	≈ 29K	16.50	0.56
1	✓	✓	Pre	≈ 38K	**18.25**	**0.60**

All variants use post-caption supervision. MIM predicts VAE latents from pre- or post-quantization features. Code-use PPL is measured on 200K images sampled uniformly across tokenizer-training sources; K denotes thousands. Dashes indicate disabled objectives or unavailable metrics.

(b) Visual-vocabulary initialization

	Without caption adaptation						With caption adaptation		
Metric	Random	Raw	Raw/ L2	L2	Text- L2	TAN w/o adapt.	Std. LN	Proj. +L2	TAN
GenEval ↑	0.15	0.19	0.27	0.35	0.34	0.34	0.13	0.36	**0.40**
Input norm	0.93	9.19	9.19	1.00	0.93	0.93	28.0	1.00	0.96
Output norm	0.93	9.19	1.00	1.00	0.93	0.93	28.0	1.00	0.96

L2 normalizes each embedding to unit norm; Text-L2 rescales it to the text reference norm $T \approx 0.93$. TAN w/o adapt. skips caption adaptation. Proj.+L2 adapts the vocabulary projector with L2-normalized outputs. Raw/L2 uses separate, untied raw input and L2 output weights; all other schemes, and Init. in (c), apply to both interfaces. Norms are mean per-token ℓ_2 norms before downstream training.

(c) Unified understanding and generation

Objectives	Init.	GenEval ↑	DPG ↑	TextVQA ↑	MMB-EN ↑
C.L.	L2	0.52	67.76	37.12	48.45
C.L. + D2C	L2	0.54	69.05	37.01	48.30
C.L. + D2C	TAN	0.58	71.72	38.80	50.31
C.L. + D2C + Pre-MIM	TAN	**0.62**	**73.15**	**40.51**	**52.09**

Component contributions under unified training. Table 3(c) retains pre- and post-caption supervision across variants and fixes the downstream data mixture and optimization settings. Adding D2C improves GenEval/DPG-Bench from 0.52/67.76 to 0.54/69.05, with little change in understanding scores. With this tokenizer fixed, replacing L2 with TAN raises generation to 0.58/71.72 and TextVQA/MMBench-EN from 37.01/48.30 to 38.80/50.31. Adding pre-MIM gives the best results across all four benchmarks.

Overall, the studies distinguish improvements to the tokenizer from improvements to vocabulary transfer: direct pre-quantization supervision improves reconstruction and code-use diversity, while initialization affects generation with a fixed tokenizer. Their combination improves both generation and understanding over the baseline in the controlled unified-training setting.

5 Conclusion

We presented **HUG**, a framework for unified image understanding and autoregressive image generation. HUG combines a shared-encoder dual-branch tokenizer with adaptation of the learned visual vocabulary to a pretrained LLM. Our controlled studies show that adding direct caption and masked VAE-latent supervision before quantization improves reconstruction and code-use diversity beyond the post-quantization objectives used in our setting. With the tokenizer fixed, vocabulary initialization also affects generation, and caption-adapted TAN outperforms the tested alternatives. The complete system achieves strong image generation and competitive understanding performance among the unified models evaluated. These findings support jointly considering tokenizer supervision and vocabulary transfer when designing unified multimodal models.

AI USE STATEMENT

LLM-based tools were used during manuscript preparation to revise prose, assist with LaTeX formatting, and review the clarity of technical explanations. We also used them to discuss possible experimental controls and assess whether the wording of our claims was supported by the reported evidence. All decisions about experimental design, interpretation, and reporting were made by the authors, who take responsibility for the content of this paper.

ETHICS STATEMENT

Unified multimodal understanding and generation can benefit creative design, accessibility, education, and human–computer interaction. However, stronger image-generation capabilities may also be misused to create misleading or harmful content, amplify data biases, or reproduce sensitive material. Responsible deployment therefore requires data governance, safety filtering, provenance mechanisms, and clear usage policies.

REPRODUCIBILITY STATEMENT

We describe HUG's model architecture, training stages, and evaluation procedures in the main text and appendix. Implementation details include the pretrained components, tokenizer objectives, vocabulary adaptation procedure, and controlled ablation protocols. Our training recipe makes extensive use of publicly available datasets, with stage-specific data sources described in Appendix A.3. It also includes in-house and self-synthesized data, whose limited availability constrains exact reproduction of the full-system results. The generation-only initialization ablations use the public BLIP-3o-30M dataset, and their configurations are provided in Appendix A.4.

REFERENCES

Yuang Ai, Jiaming Han, Shaobin Zhuang, Weijia Mao, Xuefeng Hu, Ziyan Yang, Zhenheng Yang, Yali Wang, Huaibo Huang, Xiangyu Yue, et al. Bitdance: Scaling autoregressive generative models with binary tokens. *arXiv preprint arXiv:2602.14041*, 2026.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, Chunsheng Wu, Huajie Tan, Chunyuan Li, Jing Yang, Jie Yu, Xiyao Wang, Bin Qin, Yumeng Wang, Zizhen Yan, Ziyong Feng, Ziwei Liu, Bo Li, and Jiankang Deng. Llava-onevision-1.5: Fully open framework for democratized multimodal training, 2025. URL https://arxiv.org/abs/2509.23661.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Bingyi Cao, Koert Chen, Kevis-Kokitsi Maninis, Kaifeng Chen, Arjun Karpur, Ye Xia, Sahil Dua, Tanmaya Dabral, Guangxing Han, Bohyung Han, Joshua Ainslie, Alex Bewley, Mithun Jacob, René Wagner, Washington Ramos, Krzysztof Choromanski, Mojtaba Seyedhosseini, Howard Zhou, and André Araujo. TIPSv2: Advancing Vision-Language Pretraining with Enhanced Patch-Text Alignment. In *CVPR*, 2026.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025a.

Jiuhai Chen, Le Xue, Zhiyang Xu, Xichen Pan, Shusheng Yang, Can Qin, An Yan, Honglu Zhou, Zeyuan Chen, Lifu Huang, Tianyi Zhou, Junnan Li, Silvio Savarese, Caiming Xiong, and Ran Xu. Blip3o-next: Next frontier of native image generation. *arXiv preprint arXiv:2510.15857*, 2025b. URL https://arxiv.org/abs/2510.15857.

Junying Chen, Zhenyang Cai, Pengcheng Chen, Shunian Chen, Ke Ji, Xidong Wang, Yunjin Yang, and Benyou Wang. Sharegpt-4o-image: Aligning multimodal models with gpt-4o-level image generation, 2025c.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025d.

Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, Yueze Wang, Chengyu Wang, Fan Zhang, Yingli Zhao, Ting Pan, Xianduo Li, Zecheng Hao, Wen-Rui Ma, Zhuo Chen, Yulong Ao, Tiejun Huang, Zhongyuan Wang, and Xinlong Wang. Emu3.5: Native multimodal models are world learners. *ArXiv*, abs/2510.26583, 2025. URL https://api.semanticscholar.org/CorpusID:282592521.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025a.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025b.

Nirajan Dhakal. Realworldqa dataset. https://huggingface.co/datasets/nirajandhakal/realworldqa, 2023. Hugging Face Datasets.

Haiwen Diao, Penghao Wu, Hanming Deng, Jiahao Wang, Shihao Bai, Silei Wu, Weichen Fan, Wenjie Ye, Wenwen Tong, Xiangyu Fan, et al. Sensenova-u1: Unifying multimodal understanding and generation with neo-unify architecture. *arXiv preprint arXiv:2605.12500*, 2026.

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 *Proceedings of the 41st International Conference on Machine Learning*, pp. 12606–12633, 2024.

Rongyao Fang, Aldrich Yu, Chengqi Duan, Linjiang Huang, Shuai Bai, Yuxuan Cai, Kun Wang, Si Liu, Xihui Liu, and Hongsheng Li. FLUX-reason-6m & PRISM-bench: A million-scale text-to-image reasoning dataset and comprehensive benchmark. In *The Fourteenth International Conference on Learning Representations*, 2026.

Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, et al. Seedream 3.0 technical report. *arXiv preprint arXiv:2504.11346*, 2025.

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: 52132–52152, 2023.

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*, 2025a.

NextStep Team Chunrui Han, Guopeng Li, Jingwei Wu, Quan Sun, Yan Cai, Yuang Peng, Zheng Ge, Deyu Zhou, Haomiao Tang, Hongyu Zhou, Kenkun Liu, Ailin Huang, Bin Wang, Changxing Miao, Deshan Sun, En Yu, Fukun Yin, Gang Yu, Hao Nie, Haoran Lv, Hanpeng Hu, Jia Wang, Jian Zhou, Jian-Yuan Sun, Kaijun Tan, Kang An, Kangheng Lin, Liang Zhao, Mei Chen, Peng Xing, Rui Wang, Shiyu Liu, Shu-Tao Xia, Tian You, Wei Ji, Xianfang Zeng, Xin pu Han, Xuelin Zhang, Yana Wei, Yanming Xu, Yimin Jiang, Yingming Wang, Yu Zhou, Yu Han, Zi Meng, Binxing Jiao, Daxin Jiang, Xiangyu Zhang, and Yibo Zhu. Nextstep-1: Toward autoregressive image generation with continuous tokens at scale. *ArXiv*, abs/2508.10711, 2025b. URL https://api.semanticscholar.org/CorpusID:280649872.

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.

Ziyuan Huang, DanDan Zheng, Cheng Zou, Rui Liu, Xiaolong Wang, Kaixiang Ji, Weilong Chai, Jianxin Sun, Libin Wang, Yongjie Lv, et al. Ming-univision: Joint image understanding and generation with a unified continuous tokenizer. *arXiv preprint arXiv:2510.06590*, 2025.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images, 2016.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Bozhou Li, Xinda Xue, Sihan Yang, Yang Shi, Xinlong Chen, Yushuo Guan, Yuanxing Zhang, and Wentao Zhang. The unseen bias: How norm discrepancy in pre-norm MLLMs leads to visual information loss. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Yanghao Li, Rui Qian, Bowen Pan, Haotian Zhang, Haoshuo Huang, Bowen Zhang, Jialing Tong, Haoxuan You, Xianzhi Du, Zhe Gan, Hyunjik Kim, Chao Jia, Zhenbang Wang, Yinfei Yang, Mingfei Gao, Zi-Yi Dou, Wenze Hu, Chang Gao, Dongxu Li, Philipp Dufter, Zirui Wang, Guoli Yin, Zhengdong Zhang, Chen Chen, Yang Zhao, Ruoming Pang, and Zhifeng Chen. MANZANO: A simple and scalable unified multimodal model with a hybrid vision tokenizer. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Yuanzhan 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 multimodal model an all-around player? In *European Conference on Computer Vision*, 2023a. URL https://api.semanticscholar.org/CorpusID:259837088.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xucheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67, 2023b. URL https://api.semanticscholar.org/CorpusID:274769861.

Zhiheng Liu, Weiming Ren, Haozhe Liu, Zijian Zhou, Shoufa Chen, Haonan Qiu, Xiaoke Huang, Zhaochong An, Fanny Yang, Aditya Patel, et al. Tuna: Taming unified visual representations for native unified multimodal models. *arXiv preprint arXiv:2512.02014*, 2025.

Zhiheng Liu, Weiming Ren, Xiaoke Huang, Shoufa Chen, Tianhong Li, Mengzhao Chen, Yatai Ji, Sen He, Jonas Schult, Belinda Zeng, Tao Xiang, Wenhu Chen, Ping Luo, Luke S. Zettlemoyer, and Yuren Cong. Tuna-2: Pixel embeddings beat vision encoders for multimodal understanding and generation. 2026. URL https://api.semanticscholar.org/CorpusID:287782690.

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. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026a.

Xu Ma, Yitian Zhang, Qihua Dong, and Yun Fu. Fine-t2i: An open, large-scale, and diverse dataset for high-quality t2i fine-tuning, 2026b.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq R. Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *ArXiv*, abs/2203.10244, 2022. URL https://api.semanticscholar.org/CorpusID:247593713.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mahmoud Assran, Koustuv Sinha, Michael Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026.

Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, Ji Hou, and Saining Xie. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

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. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=jwRAIYoVJn.

Qi Qin, Le Zhuo, Yi Xin, Ruoyi Du, Zhen Li, Bin Fu, Yiting Lu, Jiakang Yuan, Xinyue Li, Dongyang Liu, Xiangyang Zhu, Manyuan Zhang, Will Beddow, Erwann Millon, Victor Perez, Wen-Hao Wang, Conghui He, Bo Zhang, Xiaohong Liu, Hongsheng Li, Yu-Hao Qiao, Chang Xu, and Peng Gao. Lumina-image 2.0: a unified and efficient image generative framework. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20031–20042, 2025. URL https://api.semanticscholar.org/CorpusID:277349538.

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*, pp. 10684–10695, 2022.

Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, et al. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025.

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

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

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, Congcong Wang, Dehao Zhang, Dikang Du, Dongliang Wang, Enming Yuan, Enzhe Lu, Fang Li, Flood Sung, Guangda Wei, Guokun Lai, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haoning Wu, Haotian Yao, Haoyu Lu, Heng Wang, Hongcheng Gao, Huabin Zheng, Jiaming Li, Jianlin Su, Jianzhou Wang, Jiaqi Deng, Jiezhong Qiu, Jin Xie, Jinhong Wang, Jingyuan Liu, Junjie Yan, Kun Ouyang, Liang Chen, Lin Sui, Longhui Yu, Mengfan Dong, Mengnan Dong, Nuo Xu, Pengyu Cheng, Qizheng Gu, Runjie Zhou, Shaowei Liu, Sihan Cao, Tao Yu, Tianhui Song, Tongtong Bai, Wei Song, Weiran He, Weixiao Huang, Weixin Xu, Xiaokun Yuan, Xingcheng Yao, Xingzhe Wu, Xinxing Zu, Xinyu Zhou, Xinyuan Wang, Y. Charles, Yan Zhong, Yang Li, Yangyang Hu, Yanru Chen, Yejie Wang, Yibo Liu, Yibo Miao, Yidao Qin, Yimin Chen, Yiping Bao, Yiqin Wang, Yongsheng Kang, Yuanxin Liu, Yulun Du, Yuxin Wu, Yuzhi Wang, Yuzi Yan, Zaida Zhou, Zhaowei Li, Zhejun Jiang, Zheng Zhang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Zijia Zhao, and Ziwei Chen. Kimi-VL technical report, 2025. URL https://arxiv.org/abs/2504.07491.

Junke Wang, Xiao Wang, Jiacheng Pan, Xuefeng Hu, Feng Li, Jingxiang Sun, Chaorui Deng, Zilong Chen, Yunpeng Chen, Kaibin Tian, et al. Arm: An autoregressive large multimodal model with unified discrete representations. *arXiv preprint arXiv:2606.11188*, 2026a.

Xinlong Wang, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Zhen Li, Yuqi Wang, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Chunlei Men, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Zhongyuan Wang, and Tiejun Huang. Multimodal learning with

next-token prediction for large multimodal models. *Nature*, 650(8101):327–333, Feb 2026b. ISSN 1476-4687. doi:10.1038/s41586-025-10041-x. URL https://doi.org/10.1038/s41586-025-10041-x.

Xinlong Wang, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Zhen Li, Yuqi Wang, et al. Multimodal learning with next-token prediction for large multimodal models. *Nature*, pp. 1–7, 2026c.

Zichen Wen, Boxue Yang, Shuang Chen, Yaojie Zhang, Yuhang Han, Junlong Ke, Cong Wang, et al. Innovator-vl: A multimodal large language model for scientific discovery. *arXiv preprint arXiv:2601.19325*, 2026.

Luis Wiedmann, Orr Zohar, Amir Mahla, Xiaohan Wang, Rui Li, Thibaud Frere, Leandro von Werra, Aritra Roy Gosthipaty, and Andrés Marafioti. Finevision: Open data is all you need, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12966–12977, 2025b.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Zhonghua Wu, Qi Tao, Wentao Liu, Wei Li, and Chen Change Loy. Harmonizing visual representations for unified multimodal understanding and generation. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17739–17750, 2025c. URL https://api.semanticscholar.org/CorpusID:277435700.

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. *ArXiv*, abs/2409.04429, 2024. URL https://api.semanticscholar.org/CorpusID:272463603.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

Junyan Ye, Dongzhi Jiang, Zihao Wang, Leqi Zhu, Zhenghao Hu, Zilong Huang, Jun He, Zhiyuan Yan, Jinghua Yu, Hongsheng Li, Conghui He, and Weijia Li. Echo-4o: Harnessing the power of gpt-4o synthetic images for improved image generation. *arXiv preprint arXiv:2508.09987*, 2025. URL https://arxiv.org/abs/2508.09987.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of CVPR*, 2024.

Z.ai. GLM-Image: Auto-regressive for dense-knowledge and high-fidelity image generation. Technical blog, January 2026. URL https://z.ai/blog/glm-image.

Yi Zhang, Bolin Ni, Xin-Sheng Chen, Heng-Rui Zhang, Yongming Rao, Houwen Peng, Qinglin Lu, Han Hu, Meng-Hao Guo, and Shi-Min Hu. Bee: A high-quality corpus and full-stack suite to unlock advanced fully open mllms, 2025.

Yichen Zhang, Da Peng, Zonghao Guo, Zijian Zhang, Xuesong Yang, Tong Sun, Shichu Sun, Yidan Zhang, Yanghao Li, Haiyan Zhao, et al. Cheers: Decoupling patch details from semantic representations enables unified multimodal comprehension and generation. *arXiv preprint arXiv:2603.12793*, 2026.

Chunting Zhou, LILI YU, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *The Thirteenth International Conference on Learning Representations*, 2025.

Shaobin Zhuang, Yuang Ai, Jiaming Han, Weijia Mao, Xiaohui Li, Fangyikang Wang, Xiao Wang, Yan Li, Shanchuan Lin, Kun Xu, et al. UniWeTok: An unified binary tokenizer with codebook size 2^{128} for unified multimodal large language model. *arXiv preprint arXiv:2602.14178*, 2026.

Yeyao Ma Chen Li Yongming Rao Shuyang Gu Zhao Zhong Qinglin Lu Han Hu Xiaosong Zhang Linus Di Wang Zigang Geng, Yibing Wang and Jie Jiang. X-omni: Reinforcement learning makes discrete autoregressive image generative models great again. *CoRR*, abs/2507.22058, 2025.

Kai Zou, Zhiwei Zhao, Bin Liu, and Nenghai Yu. Advancing aesthetic image generation via composition transfer. *International Journal of Computer Vision*, 134:252, 2026. doi:10.1007/s11263-026-02862-8.

A Implementation and Experimental Details

This appendix provides the model architecture, training procedure, ablation protocols, and evaluation settings used in HUG.

A.1 Model Architecture

Language backbones. We use Qwen3-0.6B as the auxiliary LLM during visual tokenizer training. For downstream unified understanding and generation, HUG-0.6B and HUG-4B use Qwen3-0.6B and Qwen3-4B-Instruct, respectively. The two variants reuse the trained visual tokenizer with backbone-specific projectors that map visual features to the corresponding LLM hidden dimension.

Visual encoder, adapters, and quantization. We use SigLIP2-Base-Patch16-NaFlex[1] as the shared visual encoder. Both continuous and discrete adapters use a spatial merge size of 2. The discrete branch projects the merged features to a 512-dimensional pre-quantization latent. FSQ then linearly maps this latent to six scalar quantization dimensions and applies bounding and rounding with a straight-through estimator. We use a single codebook with levels $[8, 8, 8, 5, 5, 5]$, defining $64{,}000$ possible codes. The quantized codes are projected back to 512 dimensions and subsequently mapped to the target LLM's hidden dimension by the vocabulary projector. Pre-caption and pre-MIM supervise the 512-dimensional latent before the FSQ input projection. D2C compares the projected quantized features with the corresponding continuous-adapter outputs, using stop-gradient on the continuous targets.

Caption-path sampling and auxiliary losses. At each training step, we sample one visual pathway and compute a single caption cross-entropy loss. With pre-caption enabled, the continuous, pre-quantization, and post-quantization pathways are sampled with equal probability ($1/3$ each). Without pre-caption, the continuous and post-quantization pathways are sampled with equal probability ($1/2$ each). The caption loss has weight 1.0 for all pathways. In the post-caption reweighting control, this weight is increased to 2.0 only when the post-quantization pathway is selected; it remains 1.0 for the continuous pathway. D2C and MIM, when enabled, are computed at every training step independently of the sampled caption pathway, with weights of 1.0 and 0.2, respectively.

Masked VAE-latent prediction. We sample a mask ratio $r \in [0.1, 0.5]$ and randomly mask the corresponding proportion of positions in the 512-dimensional pre-quantization latent. The masked sequence is fed to an auxiliary Transformer to predict the corresponding Z-Image VAE latents. Inspired by TIPSv2 (Cao et al., 2026) and V-JEPA 2.1 (Mur-Labadia et al., 2026), we compute the MSE loss over all target positions, including both masked and unmasked locations, rather than only the masked positions.

Image decoder. We use Lumina-Image 2.0 (Qin et al., 2025) as the image decoder. Given the visual token IDs predicted by the LLM, we use their corresponding feature representations from the visual tokenizer learned in the first stage and use these features to condition image generation. The decoder does not receive the text prompt. Its visual conditioning features come from the trained tokenizer, rather than the TAN-initialized input embeddings or output weights of the downstream LLM.

Parameter counts. The 0.6B and 4B designations refer to the original language backbones, not the total HUG system.

A.2 Training Procedure and Optimization

HUG is trained through visual tokenizer learning, image-decoder training, TAN adaptation, unified pretraining, instruction tuning, and supervised fine-tuning (SFT). Table 4 summarizes the optimization settings. For the 4B backbone, we additionally adapt the continuous-branch and post-quantization projectors before TAN adaptation.

[1]Checkpoint: google/siglip2-base-patch16-naflex on Hugging Face.

Table 4: Training stages and optimization settings. LR denotes learning rate. The TAN adaptation row specifies the 0.6B configuration.

Stage	Trainable modules	Steps	LR	Batch
Tokenizer training	Visual tokenizer, auxiliary prediction heads, and Qwen3-0.6B	400K	2×10^{-5}	256
Decoder training	Lumina-Image 2.0 generation network	100K	2×10^{-5}	128
TAN adaptation	TAN and auxiliary Qwen3-0.6B	50K	2×10^{-5}	256
Unified pretraining	Target LLM	120K	1×10^{-4}	1024
Instruction tuning	Target LLM	350K	5×10^{-5}	512
SFT	Target LLM	50K	2×10^{-5}	128

Tokenizer training. We train the visual tokenizer with Qwen3-0.6B as the shared auxiliary captioning model. For each caption-supervision forward pass, we sample one of three visual representations with equal probability ($1/3$ each): continuous-branch features, pre-quantization features, or post-quantization features. The selected features are mapped to the LLM's hidden dimension through their corresponding projector and used to condition caption prediction. The loss weights are 1, 0.2, 1, and 1 for pre-caption, pre-MIM, post-caption, and D2C, respectively.

Masked VAE-latent prediction. We mask a subset of positions in the 512-dimensional pre-quantization latent before feeding it to the auxiliary Transformer. The predictor reconstructs the corresponding Z-Image VAE latents. Inspired by TIPSv2 (Cao et al., 2026) and V-JEPA 2.1 (Mur-Labadia et al., 2026), we apply the MSE loss to all target positions, including both masked and unmasked locations, rather than restricting supervision to masked positions.

D2C gradient handling. D2C compares the projected quantized features with the continuous-adapter outputs. The continuous targets are detached from this loss, so gradients do not propagate through the target path. This does not freeze the shared encoder, which can receive gradients through the discrete branch.

Image-decoder training. We freeze the visual tokenizer and train Lumina-Image 2.0 (Qin et al., 2025) to reconstruct images conditioned on their encoded visual-token features. These features are obtained from the first-stage tokenizer, not from the TAN-initialized weights of the downstream LLM. The decoder does not receive the text prompt. All parameters of the Lumina generation network are trainable in this stage, while the image VAE remains frozen.

Backbone-specific projection. For Qwen3-0.6B, we retain the projectors learned during tokenizer training. For Qwen3-4B-Instruct, we replace the continuous-branch and post-quantization projectors with newly initialized projectors matching the 4B hidden dimension. We jointly train these two projectors and the 4B LLM using caption supervision on the corresponding visual features, while keeping the remaining tokenizer components frozen. Pre-quantization caption prediction, MIM, and D2C are not applied during this projector-adaptation stage.

TAN adaptation and transfer. After projector training, we freeze the tokenizer and projector and optimize TAN together with an auxiliary LLM using caption supervision. For the 0.6B configuration, adaptation continues from the auxiliary Qwen3-0.6B used during tokenizer training. After adaptation, only the transformed visual vocabulary is transferred. The target LLM is loaded from its original pretrained checkpoint, and the transformed vocabulary initializes its visual input embeddings and output prediction weights. The adapted auxiliary LLM weights are not transferred.

Unified training. We jointly train the target LLM on understanding and generation while keeping the visual tokenizer and image decoder frozen. The understanding-to-generation sampling ratio is 1:2 during unified pretraining and instruction tuning, and 1:1 during SFT. The target LLM parameters, including the visual-token input and output weights, are optimized in these stages.

Computational resources. All training and evaluation experiments are conducted on NVIDIA H800 GPUs. The full training pipeline uses 128 H800 GPUs for approximately three weeks.

A.3 Training Data

HUG uses stage-specific data mixtures for tokenizer learning, image decoding, vocabulary adaptation, and unified understanding-and-generation training. The sampling ratios below are listed in understanding-to-generation order. Optimization settings are summarized in Table 4.

Visual tokenizer training. We train the dual-branch tokenizer for 400K steps: 200K steps on image-caption data (An et al., 2025), followed by 200K steps on FineVision (Wiedmann et al., 2025). These data support the auxiliary language-prediction objectives used during tokenizer learning.

Image-decoder training. We finetune Lumina-Image 2.0 on 3M image-text pairs for 100K steps. The images provide both the encoded visual-token conditioning and the reconstruction targets. Although the dataset consists of image-text pairs, the decoder receives only the visual-token features, not the associated text.

TAN adaptation. For the full-model recipe, TAN adaptation uses image-caption data (An et al., 2025) for 50K steps. The auxiliary LLM predicts the paired captions conditioned on the transformed visual embeddings. The separate adaptation protocol used in the initialization ablations is described in Appendix A.4.

Unified multimodal pretraining. We perform unified pretraining for 120K steps with an understanding-to-generation sampling ratio of 1:2. The understanding data consists of image-caption corpora (An et al., 2025). The generation mixture includes 50M in-house text-to-image pairs, BLIP-3o-30M (Chen et al., 2025a), FLUX-Reason-6M (Fang et al., 2026), and Text-to-Image-2M (Zou et al., 2026).

Instruction tuning. We continue joint training for approximately 350K steps, retaining the 1:2 understanding-to-generation sampling ratio. The understanding data is drawn from Innovator-VL-Instruct-46M (Wen et al., 2026). The generation mixture includes FineT2I (Ma et al., 2026b), Cheers (Zhang et al., 2026), FLUX-Reason-6M (Fang et al., 2026), and 7M self-synthesized image-generation instruction examples.

Supervised fine-tuning. We perform SFT for approximately 50K steps with a 1:1 understanding-to-generation sampling ratio. The understanding mixture consists of Cheers (Zhang et al., 2026) and Honey-1M (Zhang et al., 2025). The generation mixture includes BLIP-3o-sft, ShareGPT-4o-Image (Chen et al., 2025c), Echo-4o-Image (Ye et al., 2025), and the same 7M self-synthesized instruction examples used during instruction tuning.

A.4 Ablation Protocols

The studies in Table 3 separately evaluate tokenizer supervision, vocabulary initialization, and component contributions under unified training. All downstream ablations use Qwen3-0.6B. Table 5 summarizes the training budgets for the tokenizer and initialization studies.

Table 5: Training configurations for panels (a) and (b) of Table 3. Caption adaptation applies only to the adapted initialization variants. LR denotes learning rate.

Stage	Training data	Steps	Batch	LR
(a) Tokenizer	Image-caption data, then FineVision	400K	256	2×10^{-5}
(a) Decoder	3M image-text pairs	100K	128	2×10^{-5}
(b) Caption adaptation	Image-caption data	100K	512	5×10^{-5}
(b) Generation training	BLIP-3o-30M	200K	256	5×10^{-5}

Tokenizer supervision: panel (a). All variants use the same image encoding and quantization architecture, with matched training steps and computational budgets. Following Section 4.1, tokenizer training uses image-caption data (An et al., 2025) for 200K steps, followed by FineVision (Wiedmann

et al., 2025) for another 200K steps. Post-quantization caption supervision is retained in all variants. The comparisons add D2C, pre-quantization caption supervision, and masked VAE-latent prediction. The loss-reweighting control increases the post-caption coefficient from 1 to 2 while keeping the remaining loss weights and training settings unchanged.

For the MIM-placement comparison, both variants retain pre-caption, post-caption, and D2C supervision. Pre-MIM uses the 512-dimensional latent before the FSQ input projection. Post-MIM instead uses features after FSQ, mapped through an additional projector to form the predictor input. The two variants use the same mask ratio, masking scheme, VAE targets, and MIM loss weight. The prediction loss is computed over all target positions, including both masked and unmasked locations.

Decoder training and reconstruction. For each tokenizer variant, we separately finetune Lumina-Image 2.0 (Qin et al., 2025) for 100K steps on the same 3M image-text pairs. The decoder architecture and training configuration are matched across variants; decoder weights are not shared between runs. The visual tokenizer remains frozen. The decoder is conditioned only on the visual features associated with the encoded token IDs and does not receive the text prompt. Reconstruction evaluation uses tokens encoded from the source image, rather than tokens predicted by the autoregressive LLM. Decoder checkpoints at the same training step are evaluated under an identical reconstruction protocol.

Reconstruction evaluation. We evaluate reconstruction on all 50,000 images in the ImageNet validation set at a resolution of 1024×1024. Each image is encoded into discrete visual tokens by the corresponding tokenizer and reconstructed using its separately trained image decoder. The decoder receives only the tokenizer-derived visual features, without text conditioning or autoregressive token prediction. We report mean PSNR and SSIM over the evaluation set, using the same image preprocessing and decoder inference settings across variants.

Code-use perplexity. We sample a total of 200,000 images uniformly across the source datasets used for tokenizer training. Each tokenizer encodes this evaluation subset without parameter updates. We aggregate code counts across workers before computing the empirical distribution over the $K = 64{,}000$ FSQ codes:

$$p_k = \frac{n_k}{\sum_{j=1}^{K} n_j}, \qquad \mathrm{PPL}_{\mathrm{code}} = \exp\left(- \sum_{k:p_k>0} p_k \log p_k \right), \tag{11}$$

where n_k is the number of occurrences of code k across all encoded token positions. This is a one-pass encoding statistic, not a rolling training-window statistic. Code-use perplexity measures the effective diversity of the observed code distribution, rather than the number of distinct codes. Because the images are sampled from training sources, this analysis is not a held-out generalization evaluation.

Vocabulary initialization: panel (b) All variants use the full tokenizer from the last row of Table 3(a) and the same trained image decoder. The image-tokenization and decoding pathways remain unchanged. Each variant initializes a fresh copy of the same pretrained Qwen3-0.6B checkpoint and undergoes generation-only training on BLIP-3o-30M (Chen et al., 2025a) for 200K steps, with a global batch size of 256 and a learning rate of 5×10^{-5}.

Input-output parameterization. Table 6 distinguishes the initialization values, downstream trainability, and parameter sharing of the visual input and output weights. Weight tying means that input lookup and output prediction use the same parameter tensor, not merely equal initial values.

For Raw/L2, visual inputs use a separate embedding table initialized from the raw tokenizer vocabulary. The visual-token output weights are initialized from the L2-normalized vocabulary. The visual input and output weights are therefore untied in this configuration. This is distinct from the tied input-output parameterization used in the final HUG models.

Supplementary training curves. The Raw, L2, and TAN curves in Appendix A.6 use BLIP-3o-30M, 200K training steps, and a global batch size of 256. This separate study uses a learning rate of 1.5×10^{-4} rather than the 5×10^{-5} used in panel (b). The three runs share the same tokenizer, decoder, training data, and optimization schedule.

Unified training: panel (c) All variants use both pre- and post-quantization caption supervision during tokenizer learning. The study successively adds D2C, replaces L2 initialization with TAN, and adds pre-MIM. Downstream understanding and generation are trained jointly using the same data mixture and optimization settings across variants, with a learning rate of 1.5×10^{-4}.

Table 6: Visual input/output parameterization for the vocabulary-initialization ablation. Trainability refers to downstream text-to-image training, not the preceding caption-adaptation stage. Tied weights share a parameter tensor throughout downstream training.

Variant	Input initialization	Output initialization	Input trained?	Output trained?	Tied?
Random	Random	Random	Y	Y	Y
Raw	Raw vocabulary	Raw vocabulary	Y	Y	Y
Raw/L2	Separate raw visual table	L2-normalized vocabulary	Y	Y	No
L2	Unit-normalized vocabulary	Unit-normalized vocabulary	Y	Y	Y
Text-L2	L2 scaled by T	L2 scaled by T	Y	Y	Y
TAN w/o adapt.	Fixed text-scale LN	Fixed text-scale LN	Y	Y	Y
Std. LN	Adapted LN vocabulary	Adapted LN vocabulary	Y	Y	Y
Proj.+L2	Adapted projector + L2	Adapted projector + L2	Y	Y	Y
TAN	Adapted TAN vocabulary	Adapted TAN vocabulary	Y	Y	Y

Fixed initialization strategies. Let $\mathbf{e}_k$ denote the original vocabulary embedding for visual token k, and let T be the mean norm of the target LLM's pretrained text input embeddings. Raw initialization transfers $\mathbf{e}_k$ directly. L2 initialization applies `F.normalize(..., p=2, dim=-1)` independently to each vocabulary embedding. Text-L2 additionally scales the normalized embedding by T:

$$\mathbf{e}_k^{\text{L2}} = \frac{\mathbf{e}_k}{\|\mathbf{e}_k\|_2}, \qquad \mathbf{e}_k^{\text{Text-L2}} = T\frac{\mathbf{e}_k}{\|\mathbf{e}_k\|_2}. \tag{12}$$

TAN without adaptation applies the text-scale-aware LayerNorm transformation with initial gain $(T/\sqrt{D})\mathbf{1}$ and zero bias, but skips caption adaptation. It still undergoes the same downstream generation training as the other variants. Normalization is used to construct the initial weights and is not enforced during downstream training.

For Raw/L2, visual inputs use a separate embedding table initialized from the raw tokenizer vocabulary, while the visual-token output weights are initialized from its L2-normalized counterpart. The visual input and output weights are therefore untied in this configuration.

Caption-adapted initialization strategies. Standard LayerNorm, Proj.+L2, and TAN use an additional adaptation stage on image-caption data. This stage runs for 100K steps with a global batch size of 512 and a learning rate of 5×10^{-5}. The auxiliary LLM predicts the paired caption conditioned on transformed visual embeddings. All three variants use the same adaptation data, training budget, optimization settings, and auxiliary LLM update policy.

Standard LayerNorm and TAN optimize their affine parameters. The former starts with unit gain and zero bias, whereas TAN starts with gain $(T/\sqrt{D})\mathbf{1}$ and zero bias. Proj.+L2 instead optimizes the final vocabulary projector with unit-normalized outputs. Its adapted projector is used only to construct the transferred embeddings; it does not replace the projector in the downstream image tokenizer or decoder-conditioning pathway. The remaining tokenizer parameters are frozen. After adaptation, only the resulting vocabulary embeddings are transferred to the freshly loaded target LLM. The adapted auxiliary LLM weights are not transferred.

A.5 Evaluation and Checkpoint Selection

For multimodal understanding, we evaluate HUG on MMStar (Chen et al., 2024), MMBench-EN Liu et al. (2023a), OCRBench Liu et al. (2023b), ChartQA Masry et al. (2022), AI2D (Kembhavi et al., 2016), MMMU Yue et al. (2024), and RealWorldQA (Dhakal, 2023). We follow the standard evaluation protocol of each benchmark and report the official accuracy or score. All images are

processed by the continuous visual branch, and the model generates textual answers autoregressively without using the discrete image-generation branch.

For image generation, we evaluate on GenEval Ghosh et al. (2023) and DPG-Bench Hu et al. (2024). Given each text prompt, HUG first predicts a sequence of discrete visual token IDs with the autoregressive LLM. The corresponding tokenizer-derived features then condition the Lumina-Image 2.0 decoder to generate an image.

A.6 Training Dynamics of Vocabulary Initialization

We compare Raw, L2, and TAN initialization in a supplementary generation-only training study. All three runs use the same tokenizer and image decoder, Qwen3-0.6B as the language backbone, and BLIP-3o-30M for 200K training steps with a global batch size of 256. The learning rate is 1.5×10^{-4}, compared with 5×10^{-5} in Table 3(b). As shown in Figure 4, TAN reaches a lower training loss toward the end of the evaluated budget.

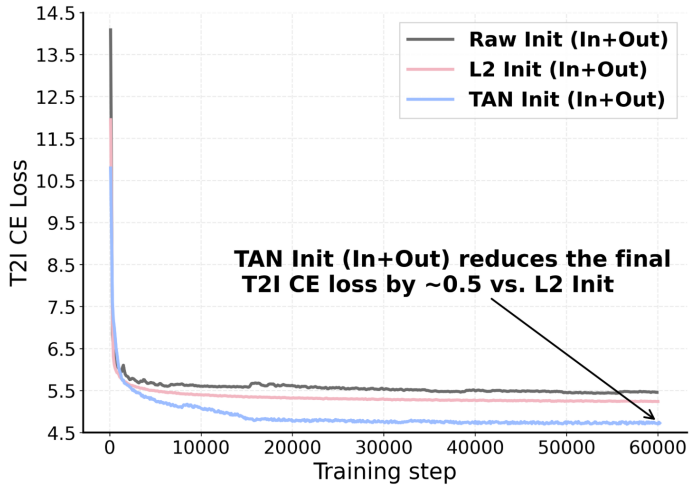


Figure 4: Generation-only training loss for Raw, L2, and TAN initialization. All three runs use Qwen3-0.6B and BLIP-3o-30M, with a global batch size of 256, 200K training steps, and a learning rate of 1.5×10^{-4}.

B Limitations and Future Work

HUG has several limitations. The main results compare complete systems whose training data, computational budgets, and image decoders differ; they should not be interpreted as matched causal comparisons. Our controlled ablations use a 0.6B backbone and simplified training settings rather than repeating every component removal throughout the full training pipeline. The system also relies on internal data and a pretrained image decoder, which limit direct reproduction of the headline results. Although HUG achieves competitive understanding among the unified models evaluated, it does not establish parity with dedicated understanding-only models. Extending the evaluation to image editing, longer multimodal interactions, and broader application domains remains future work.