GSTAR: GROUP-SCALED TOKEN CAPACITY FOR DEEPLY COMPRESSED AUTOREGRESSIVE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete autoregressive models offer a scalable framework for image generation. However, they face a fundamental trade-off: long token sequences preserve representation quality but slow generation, while aggressive spatial compression reduces representation capacity. Based on product quantization, we show that, under a fixed nominal information budget, group size should scale quadratically with the per-axis spatial compression factor. Building on this principle, we propose GSTAR, a spatial–group co-scaling framework that enables $32\times$ and even effective $64\times$ compression while preserving strong reconstruction, understanding, and generation performance. Since larger groups introduce intra-group dependencies that must be modeled efficiently, we analyze their impact on inference latency and generation quality and introduce a semi-autoregressive decoder that combines parallel feature computation with lightweight causal modeling. With these designs, our highly compressed models perform competitively with leading models across five representative text-to-image benchmarks (GenEval, DPG-Bench, PRISM, LongText-Bench, and Qwen-Image-Bench). At 1024×1024 resolution, our model delivers a $20\times$ speedup over prior discrete autoregressive models, generating each image in 2.5 seconds on a single NVIDIA H200. Together, these results show that highly compressed discrete representations can enable high-quality autoregressive generation at low latency.

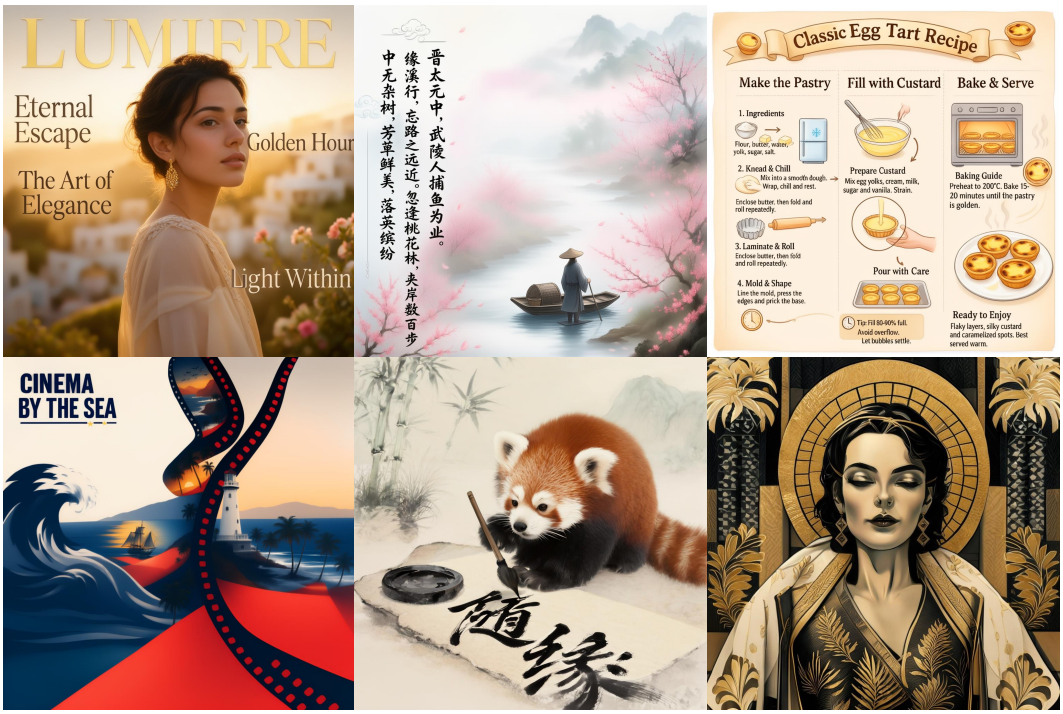
1 INTRODUCTION

Discrete visual tokenization provides a natural interface between image generation and autoregressive sequence modeling. By representing images as sequences of discrete symbols, visual generation can be formulated using the same next-token prediction paradigm that has driven the success of large language models (Esser et al., 2021; Sun et al., 2024). This formulation has also shown promise for unifying visual generation and multimodal understanding within a single autoregressive architecture (Ma et al., 2025; Zhang et al., 2026).

Despite its conceptual simplicity and scalability, discrete visual autoregressive modeling faces two fundamental limitations. First, mapping continuous visual features to a finite codebook introduces quantization error, which can discard fine-grained details and degrade reconstruction fidelity (van den Oord et al., 2017; Esser et al., 2021). The same discretization can also remove semantically relevant cues that are important for visual understanding and language alignment, reducing the informativeness of the resulting discrete representation (Ma et al., 2025; Du et al., 2026; Zhang et al., 2026). Second, discrete autoregressive models generate visual tokens sequentially, making inference latency grow with sequence length (Sun et al., 2024; Wang et al., 2025; Ai et al., 2026; Zhang et al., 2026). This bottleneck is amplified at high resolutions, where the number of tokens grows quadratically with image side length, widening the latency gap with diffusion models that update all spatial positions in parallel at each denoising step (Ho et al., 2020; Rombach et al., 2022).

Increasing the spatial compression rate offers a direct way to reduce the number of global autoregressive steps. However, aggressive spatial compression forces each spatial token to encode more information (Chen et al., 2025a;b), exacerbating the information loss introduced by discrete quantization (van den Oord et al., 2017; Esser et al., 2021). To mitigate this loss, we adopt multi-codebook

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.

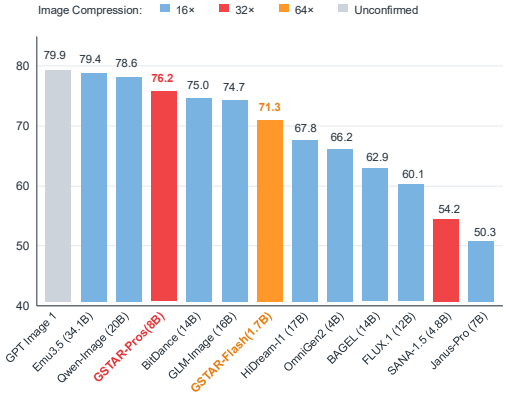


(a) Text to image generation samples from GSTAR.

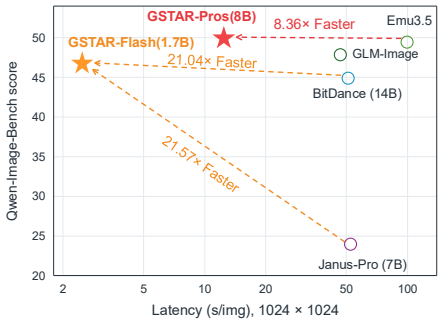


original edited original edited

(b) Image editing samples from GSTAR.



(c) Average benchmark score.



(d) Quality vs. latency.

Figure 1: Overview of GSTAR. (a) Text-to-image generation samples spanning portraits, landscapes, typography, and structured layouts. (b) Image editing examples: given an original image and a text instruction, our model produces faithful edits while preserving unrelated content (original → edited). (c) Average benchmark score comparison, where our GSTAR-Pro-8B achieves competitive performance against leading proprietary, diffusion, and discrete autoregressive models. (d) Quality vs. latency at 1024×1024 resolution, where our GSTAR-Pro-8B and GSTAR-Flash-1.7B models achieve competitive quality with significantly lower latency.

quantization (MCQ) (Lee et al., 2022; Xie et al., 2026; Ma et al., 2025), which represents each spatial token with multiple discrete codes and thereby expands its representation capacity. More importantly, MCQ admits a useful scaling property: as detailed in Section 4.2, jointly scaling the spatial compression rate and the number of codes per token can preserve the total nominal information budget while reducing the number of global autoregressive steps. We systematically study how code scaling affects reconstruction, generation, and semantic representation, and find that it enables high-compression tokenizers at $32\times$ and even $64\times$—well beyond the common $16\times$ setting—while retaining high fidelity and providing an efficient interface for downstream generation.

Multi-codebook quantization requires each global autoregressive step to predict a group of codes. RQ-Transformer-style causal depth decoders model their dependencies, but their latency grows linearly with group size and can dominate the global backbone under aggressive compression (Lee et al., 2022). Parallel and masked-token predictors remove this serial cost (Chang et al., 2022; Wang et al., 2025; Nie et al., 2025; Li et al., 2026b; Chen et al., 2026a), yet sample the codes independently and may combine incompatible modes—a failure known as multi-modal collision (Cheng et al., 2026). We therefore introduce a lightweight semi-autoregressive depth decoder: most layers run in parallel, while only the final layer retains a causal path for intra-group conditioning. This keeps group-scaling overhead nearly flat without sacrificing generation quality.

We evaluate our approach on large-scale text-to-image generation using five representative benchmarks: GenEval, DPG-Bench, PRISM, LongText-Bench, and Qwen-Image-Bench. Our models perform competitively with leading systems across these benchmarks while supporting spatial compression ratios from $32\times$ to $64\times$. Controlled studies further examine how group scaling affects reconstruction fidelity, generation quality, and semantic representation, and compare the quality–latency trade-off among causal, parallel, and semi-autoregressive intra-group decoders. At 1024×1024 resolution, our model delivers a $20\times$ speedup over prior discrete autoregressive models and generates each image in 2.5 seconds on a single NVIDIA H200.

Our main contributions are summarized as follows:

- We systematically study how scaling the group size affects reconstruction, generation, and semantic understanding under $32\times$ and $64\times$ spatial compression, showing that scaling the per-step capacity preserves strong representation quality with a short global sequence.
- We propose a lightweight semi-autoregressive depth decoder for efficiently modeling intra-group dependencies, together with an optimized inference implementation. Our system generates a 1024×1024 image in **2.5 seconds** on a single NVIDIA H200.
- We further validate the effectiveness of our approach at $32\times$ and even effective $64\times$ spatial compression by training two models, **GSTAR-Pro-8B** and **GSTAR-Flash-1.7B**, respectively, in a large-scale text-to-image setting. Our models achieve performance competitive with leading models across Qwen-Image-Bench, PRISM, DPG-Bench, GenEval, and LongText-Bench, while our approach delivers a $20\times$ inference speedup over prior discrete autoregressive models.

2 Related Work

2.1 Discrete Image Generation

Discrete generators learn distributions over quantized visual tokens (van den Oord et al., 2017; Esser et al., 2021; Sun et al., 2024). Recent advances include next-scale prediction in VAR, Infinity, and NextFlow (Tian et al., 2024; Han et al., 2025; Zhang et al., 2026), randomized generation orders in RAR and RandAR (Yu et al., 2025; Pang et al., 2025), and parallel decoding in PAR and BitDance (Wang et al., 2025; Ai et al., 2026). Janus-Pro and Emu3.5 scale discrete generation within unified multimodal systems (Chen et al., 2025c; Cui et al., 2025); UniAR further combines a shared visual tokenizer with parallel bitwise prediction (Peng et al., 2026). GLM-Image pairs autoregressive visual-token prediction with a diffusion image decoder (Z.ai Team, 2026).

Representation capacity is another key axis. RQ-Transformer, MAGVIT-v2, GSQ, UniTok, and WeTok explore residual, binary, or grouped quantization (Lee et al., 2022; Yu et al., 2024; Wang et al., 2024; Ma et al., 2025; Zhuang et al., 2026), while BAR and BitDance model large binary token spaces with masked-bit and diffusion heads, respectively (Yu et al., 2026; Ai et al., 2026). Building

on these advances, we study spatial–group co-scaling at $32\times$ and effective $64\times$ compression, jointly examining reconstruction, semantic representation, and the trade-off of group decoding.

2.2 Discrete Diffusion for Accelerated Generation

MaskGIT and Muse accelerate image generation through iterative parallel unmasking (Chang et al., 2022; 2023). Show-o combines discrete diffusion with autoregressive language modeling (Xie et al., 2025), while Emu3.5 adapts pretrained autoregressive models to parallel image-token prediction via DiDA (Cui et al., 2025). NVIDIA's Nemotron-Labs-Diffusion-Image advances this direction with token editing and grouped cross-entropy for large-vocabulary image synthesis (Li et al., 2026b).

For speculative language decoding, DFlash uses block-diffusion drafting (Chen et al., 2026a), whereas DSpark combines a parallel backbone with a lightweight sequential module to model intra-block dependencies (Cheng et al., 2026). Inspired by these approaches, we accelerate intra-group code generation by computing representations for all group slots in parallel, while using a lightweight single-layer causal Transformer to model dependencies among codes.

3 Preliminaries

Multi-codebook quantization. Multi-codebook quantization (MCQ) represents each spatial token with multiple discrete codes. We instantiate MCQ with product quantization (PQ), implemented as group-wise quantization, for its simplicity and favorable scaling: increasing the number of groups consistently improves reconstruction fidelity (Sec. 5.2.1) (Jégou et al., 2011; Wang et al., 2024; Ma et al., 2025; Zhuang et al., 2026). Specifically, PQ partitions the projected feature $\mathbf{z}_t \in \mathbb{R}^C$ into G_q disjoint channel groups and quantizes each group independently:

$$k_t^{(g)} = \arg\min_{k\in[K]} \left\| \mathbf{z}_t^{(g)} - \mathbf{e}_k^{(g)} \right\|_2^2, \qquad \hat{\mathbf{z}}_t = \mathrm{Concat}_{g=1}^{G_q} \mathbf{e}_{k_t^{(g)}}^{(g)}. \tag{1}$$

The resulting code tuple provides up to K^{G_q} configurations per spatial position without increasing the spatial sequence length.

Depth transformer. RQ-Transformer separates the generation of residual codes into a global spatial transformer and a local depth transformer (Lee et al., 2022). The global transformer summarizes preceding spatial units into a context vector, while the depth transformer models the dependencies among the codes assigned to the current unit. We adopt this spatial–depth decomposition for grouped rather than residual codes; the alternative depth-modeling strategies are detailed in Sec. 4.3.

4 Method

Our framework combines a deeply compressed discrete tokenizer with a hierarchical autoregressive generator. We use group scaling to increase the capacity of each spatial token and develop a spatial–group co-scaling principle that preserves the nominal information budget while shortening the global sequence. Building on the native $32\times$ tokenizer, we instantiate this principle through lossless spatial packing to construct an effective $64\times$ tokenizer. A lightweight semi-autoregressive depth decoder then efficiently models the correlated codes within each global step.

4.1 Deeply Compressed Tokenizer for Efficient Generation

We train a discrete tokenizer that maps an image $\mathbf{x}$ to a compact grid of group-wise codes and reconstructs it as $\hat{\mathbf{x}}$, as illustrated in Fig. 2(a). Following recent visual autoencoders and tokenizers built on pretrained semantic encoders (Yao et al., 2025; Du et al., 2026; Ma et al., 2026; Zheng et al., 2026), we initialize the encoder of our $32\times$ tokenizer from Qwen-ViT (Bai et al., 2025). The tokenizer is optimized with standard reconstruction and vector-quantization objectives. To preserve the semantic structure of the pretrained representation after quantization, we additionally align the quantized features with the corresponding Qwen-ViT features. Following Scale-RAE for text-to-image generation (Tong et al., 2026), we further employ a Gram loss (Gatys et al., 2015) to match

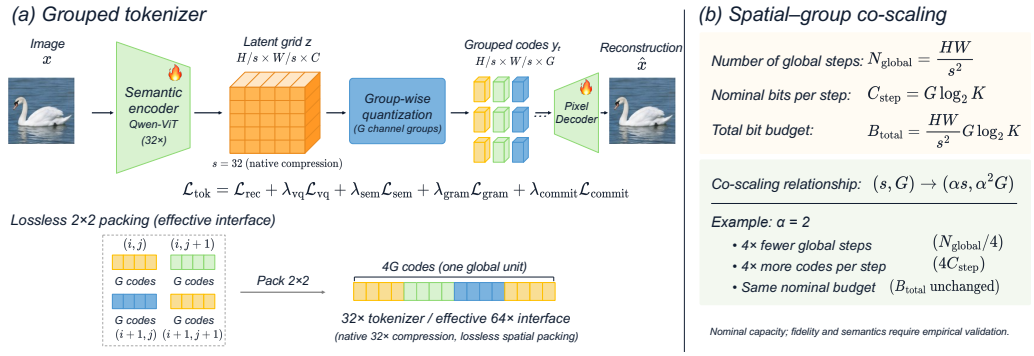


Figure 2: Grouped tokenization and spatial–group co-scaling. (a) A semantic encoder maps an image to a spatially compressed latent grid, and group-wise quantization represents each position with G discrete codes. Lossless 2 × 2 packing converts the native 32× grid into an effective 64× interface, reducing the number of global autoregressive steps by four while increasing the codes per step from G to 4G. (b) Jointly scaling spatial compression and group size preserves the nominal information capacity per image while shortening the global sequence.

second-order feature correlations between the input and its reconstruction. The overall training objective is

$$\mathcal{L}_{\text{tok}} = \mathcal{L}_{\text{rec}} + \lambda_{\text{vq}}\mathcal{L}_{\text{vq}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \lambda_{\text{gram}}\mathcal{L}_{\text{gram}} + \lambda_{\text{commit}}\mathcal{L}_{\text{commit}}. \tag{2}$$

Here $\mathcal{L}_{\text{rec}}$ collects the standard pixel, perceptual, and adversarial reconstruction terms, $\mathcal{L}_{\text{vq}}$ is the quantization objective, and $\mathcal{L}_{\text{sem}}$ is the semantic alignment objective. The commitment loss $\mathcal{L}_{\text{commit}}$ encourages encoder features to remain close to their assigned codebook embeddings.

4.2 Scaling the Group Size

Group-wise quantization provides a direct way to increase the discrete capacity assigned to each spatial token without lengthening the global sequence. This is particularly useful under aggressive spatial compression, where each remaining position must preserve more visual information. Let's denote the effective per-axis spatial compression factor and G the number of codes generated at each global position. For the native $32\times$ tokenizer, $G = G_q$, the number of quantization groups. With a fixed codebook size K, the relationships illustrated in Fig. 2(b) are

$$\begin{aligned} N_{\text{global}}(s) &= \tfrac{HW}{s^2}, \quad C_{\text{step}}(G, K) = G \log_2 K, \\ B_{\text{total}}(s, G, K) &= N_{\text{global}}\, C_{\text{step}} = \tfrac{HW}{s^2}\, G \log_2 K. \end{aligned} \tag{3}$$

Here B_{total} is a fixed-length nominal bit budget rather than the empirical entropy or effective semantic information of the learned codes.

The two variables control complementary properties. At fixed s, increasing G enlarges the representation capacity without increasing the number of global autoregressive steps. Increasing s shortens the global sequence quadratically. These properties motivate spatial–group co-scaling: scaling s by α and G by α^2 preserves B_{total} while reducing the number of global steps by a factor of α^2.

Constructing the effective $64\times$ tokenizer. Guided by this relation, we construct an effective $64\times$ tokenizer from the native $32\times$ tokenizer through the 2×2 spatial packing shown in Fig. 2(a). Each neighborhood of four quantized positions is packed into one global unit, increasing the codes per step from G_q to $4G_q$ while shortening the global sequence by a factor of four. This realizes the co-scaling relation with $\alpha = 2$, preserving all quantized codes without introducing additional quantization. The packed codes are restored to the native grid before image decoding.

To validate this capacity analysis, we systematically vary the group size and evaluate reconstruction fidelity, semantic representation, and downstream generation in Sec. 5.2.1. The results confirm that increasing group size provides additional usable information capacity and improves representation quality. Larger code tuples, however, also increase the cost of intra-group dependency modeling, motivating the efficient depth decoder introduced next.

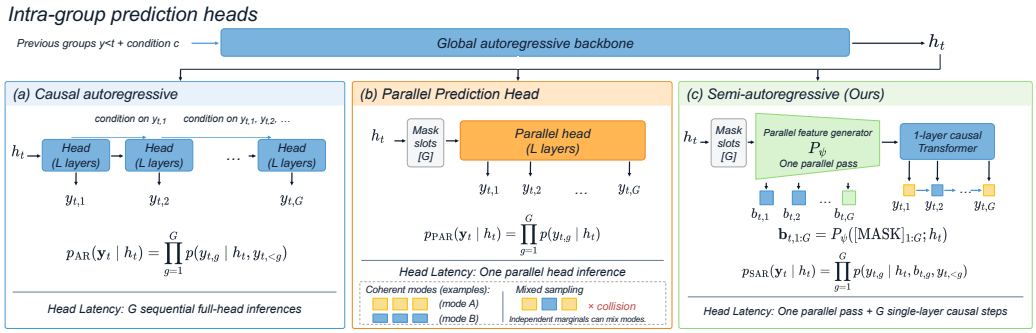


Figure 3: Comparison of intra-group prediction heads. Given the backbone context ht, (a) causal autoregressive decoding captures dependencies among codes but requires repeated, expensive head evaluations, while (b) parallel prediction head is fast but ignores these dependencies, risking inconsistent code combinations. (c) Our semi-autoregressive head addresses both limitations: a parallel feature generator performs the heavy computation once, and a lightweight single-layer causal Transformer models dependencies during sequential sampling.

4.3 Efficient Intra-Group Modeling

Group scaling requires each global autoregressive step to generate a tuple of correlated discrete codes. A common approach, following RQ-Transformer (Lee et al., 2022), is to use a causal depth head to model intra-group dependencies, but its sequential decoding incurs increasing inference latency as the group size grows. Fully parallel prediction removes this serial overhead, yet sacrifices generation quality by ignoring these dependencies. To reconcile these competing objectives, we propose a semi-autoregressive head that preserves causal dependency modeling while approaching the decoding speed of parallel prediction. Below, we first examine the causal and parallel alternatives, then present our design.

At global decoding step t, the autoregressive backbone produces a context representation $\mathbf{h}_t$. The intra-group head uses this context to generate G codes, $\mathbf{y}_t = (y_{t,1}, \ldots, y_{t,G})$, by modeling the conditional joint distribution $p(\mathbf{y}_t \mid \mathbf{h}_t)$.

Causal Autoregressive Head. Following the spatial–depth decomposition of RQ-Transformer and its use in grouped-code generation (Lee et al., 2022; Ma et al., 2025), a causal autoregressive head predicts the codes one by one, conditioning each prediction on the global context and previously generated codes:

$$p_{\mathrm{AR}}(\mathbf{y}_t \mid \mathbf{h}_t) = \prod_{g=1}^{G} p(y_{t,g} \mid \mathbf{h}_t, \mathbf{y}_{t,<g}). \tag{4}$$

This factorization preserves intra-group dependencies, but requires running the full head for each of the G sequential predictions. As G grows, this serial computation becomes a substantial decoding bottleneck and can dominate the global backbone under aggressive compression.

Parallel Prediction Head. To remove this sequential overhead, a fully parallel head predicts all G codes in a single forward pass:

$$p_{\mathrm{PAR}}(\mathbf{y}_t \mid \mathbf{h}_t) = \prod_{g=1}^{G} p(y_{t,g} \mid \mathbf{h}_t). \tag{5}$$

This removes the intra-group serial path (Chang et al., 2022; Wang et al., 2025; Nie et al., 2025; Li et al., 2026b), but makes the codes conditionally independent during sampling. Since codes within a group jointly describe the same visual content, independently sampled marginals may select incompatible modes, leading to *multimodal collision* (Cheng et al., 2026). The quality degradation motivates a head that retains causal conditioning while keeping most computation parallel.

Semi-Autoregressive Head. We propose a semi-autoregressive head that combines a parallel trunk with a lightweight causal module, retaining intra-group dependencies without repeating expensive representation modeling G times. Drawing architectural inspiration from the parallel–sequential drafters in DFlash and DSpark (Chen et al., 2026a; Cheng et al., 2026), the parallel trunk first computes representations for all group slots in one pass:

$$\mathbf{b}_{t,1:G} = P_\psi([\textsc{Mask}]_{1:G}; \mathbf{h}_t). \tag{6}$$

A lightweight, single-layer causal Transformer then generates each code conditioned on both its parallel representation and the previously sampled intra-group codes:

$$p_{\mathrm{SAR}}(\mathbf{y}_t \mid \mathbf{h}_t) = \prod_{g=1}^{G} p(y_{t,g} \mid \mathbf{h}_t, \mathbf{b}_{t,g}, \mathbf{y}_{t,<g}). \tag{7}$$

The parallel trunk is evaluated only once per global position. Only the single-layer, KV-cached causal Transformer remains sequential, allowing each code to depend on previously sampled codes without repeatedly executing the full head. Experiments in Sec. 5.2.2 show that our head achieves generation quality comparable to the fully causal head with latency close to parallel prediction, while consistently outperforming the parallel head across all three quality metrics.

5 Experiments

Experimental setup. We develop two models: **GSTAR-Pro-8B**, which uses a $32\times$ tokenizer with group size $G = 16$, and **GSTAR-Flash-1.7B**, which uses an effective $64\times$ tokenizer with $G = 32$ for fast image generation. Following prior progressive-training recipes (Deng et al., 2025; Chen et al., 2026c;b), both models are trained through pretraining, continued training (CT), and supervised fine-tuning (SFT), with progressively increasing image resolution.Detailed tokenizer training procedures and stage-wise generator training configurations are provided in Appendix A. Both models support image generation and editing, evaluated in Sec. 5.1. We further analyze the effects of group scaling on reconstruction, generation, and semantic understanding in Sec. 5.2.1, compare the quality and latency of different prediction heads in Sec. 5.2.2, and investigate the sources of acceleration in Sec. 5.3.

5.1 Main Results

We adopt GenEval (Ghosh et al., 2023), DPG-Bench (Hu et al., 2024), PRISM (Fang et al., 2026), LongText-Bench (Geng et al., 2025), and Qwen-Image-Bench (Li et al., 2026a) to evaluate text-to-image generation, covering compositional alignment, perceptual quality, text rendering, and broader prompt-following capabilities. We evaluate instruction-guided image editing on GEdit-Bench (Liu et al., 2025), reporting semantic consistency, perceptual quality, and overall scores for its English subset. Results for GSTAR-Pro-8B and GSTAR-Flash-1.7B are reported in Tables 1 and 5, respectively for generation and editing. Benchmark versions, sampling configurations, and evaluation protocols are detailed in Appendix C. We also evaluate the reconstruction quality of our visual tokenizers in Appendix E.

5.2 Ablation Studies

5.2.1 Group Scaling

We vary the group size $G \in \{1, 8, 16, 32\}$ while fixing the codebook size at $K = 16{,}384$ and the spatial compression factor at $32\times$, and examine how increased per-position capacity affects reconstruction, generative modeling, and semantic understanding. Training details for the ImageNet ablations are provided in Appendix A.3. Generally, increasing G improves reconstruction and semantic representation (Table 2), but slows early-stage generation convergence, consistent with VA-VAE (Yao et al., 2025). Suprisingly, $G = 16$ eventually catches up with and outperforms $G = 8$, despite converging more slowly at first (Fig. 4).

Table 2: Effects of group size with $K = 16{,}384$.

G	rFID ↓	gFID ↓	Linear probe ↑ (%)
1	2.0624	**2.1057**	81.45
8	0.8525	6.3773	82.90
16	0.5357	5.7388	84.15
32	**0.4325**	17.05	**84.55**

Table 1: Text-to-image generation results. PRISM is the English overall combined score (Qwen2.5-VL-72B judge), rescaled from 0–100 to 0–10. LongText reports the English subset; Qwen-Image denotes Qwen-Image-Bench overall (0–100).

Model	Comp.	Params.	GenEval ↑	DPG ↑	PRISM ↑	LongText ↑	Qwen-Image ↑
Proprietary Models							
Kolors 2.0	–	–	–	–	–	0.258	–
GPT Image 1	–	–	0.84	85.15	8.07	0.956	54.07
Seedream 3.0	–	–	0.84	88.27	7.62	0.896	–
Diffusion Models							
HiDream-I1-Full	16×	17B	0.83	85.89	7.26	0.543	43.08
FLUX.1 [dev]	16×	12B	0.66	83.84	6.85	0.607	21.58
BAGEL	16×	14B	0.88	85.07	6.20	0.373	42.04
Z-Image	16×	6B	0.84	88.14	7.56	0.935	49.10
Qwen-Image	16×	20B	0.87	88.32	7.41	0.943	49.23
OmniGen2	16×	4B	0.86	83.57	6.48	0.561	40.70
SANA-1.5	32×	4.8B	0.81	84.70	6.51	0.050	35.14
Autoregressive Models							
Janus-Pro-7B	16×	7B	0.80	84.19	6.21	0.019	23.53
GLM-Image	16×	16B	0.72	84.78	7.32	0.952	48.19
Emu3.5	16×	34.1B	0.86	88.26	7.52	0.976	49.89
BitDance	16×	14B	0.86	88.28	7.19	0.837	45.19
(Ours)GSTAR-Pro-8B	**32×**	**8B**	0.84	83.38	6.98	0.935	50.07
(Ours)GSTAR-Flash-1.7B	**64×**	**1.7B**	0.76	80.18	6.81	0.853	46.88

We further validate this observation on text-to-image generation using the same setting as the head ablation in Sec. 5.2.2: Qwen3-1.7B initialization, a $32\times$ tokenizer, 100k pretraining steps at 512×512 resolution, a sequence length of 8,192, and a global batch size of 32, followed by one epoch of SFT. Table 3 shows that $G = 16$ achieves comparable or slightly better performance.

Table 3: Text-to-image group scaling. PRISM: combined score (0–10).

G	GenEval ↑	DPG ↑	PRISM ↑
8	0.8060	82.49	5.30
16	0.8386	82.48	5.46

Together with the late-stage ImageNet results, these findings suggest that larger groups provide a richer representational space that preserves more visual and semantic information. Although the resulting space is more difficult to model and leads to slower early-stage convergence, sufficient training allows the generator to exploit its increased capacity. Larger models may benefit even further from this additional capacity.

We also validate that our lossless packing strategy, which transforms $\mathrm{f32g8}$ into $\mathrm{f64g32}$, retains strong text-to-image generation performance; see Appendix D.

5.2.2 Intra-Group Prediction Heads

We compare the Causal AR Head, Parallel Prediction Head, and our Semi-AR Head using a Qwen3-1.7B-initialized model with a $32\times$ tokenizer and $G = 16$. All variants share the same training setting: 100k pretraining steps at a fixed 512×512 resolution, a sequence length of 8,192, and a global batch size of 32, followed by one epoch of SFT.

As shown in Table 4, our Semi-AR Head substantially reduces inference latency while maintaining generation quality comparable to the Causal AR Head. It also consistently outperforms the Parallel Prediction Head across all three quality metrics.

5.3 Efficiency Analysis

Our framework enables highly compressed visual representations that shorten the global autoregressive sequence, while the Semi-AR Head reduces the latency of intra-group modeling. Together, these designs reduce decoding costs at both the global and local levels. Beyond the architectural

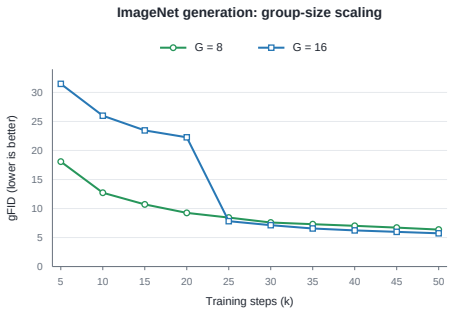


Figure 4: ImageNet gFID during training. $G = 16$ converges more slowly at first but outperforms $G = 8$ from 25k steps.

Table 4: Prediction heads with matched training. PRISM: combined score; latency is measured per image at 512×512.

Head	GenEval ↑	DPG ↑	PRISM ↑	Head ↓ (s)
Causal AR	**0.8386**	82.48	**5.46**	6.44
Parallel	0.7800	79.98	4.72	**1.77**
Semi-AR (ours)	0.8253	**82.60**	**5.46**	3.41

Table 5: Image editing on GEdit-Bench (English full set).

Model	SC ↑	PQ ↑	Overall ↑
Proprietary Models			
Gemini 2.0	6.73	6.61	6.32
Gemini 2.5 Flash Image	7.41	7.96	7.10
GPT-Image-1 [High]	7.85	7.62	7.53
Open Models			
AnyEdit	3.18	5.82	3.21
Instruct-Pix2Pix	3.58	5.49	3.68
MagicBrush	4.68	5.66	4.52
UniWorld-v1	4.93	7.43	4.85
OmniGen	5.96	5.89	5.06
FLUX.1 Kontext [Dev]	6.52	7.38	6.00
OmniGen2	7.16	6.77	6.41
BAGEL	7.36	6.83	6.52
Step1X-Edit	7.66	7.35	6.97
Qwen-Image-Edit	8.00	7.86	7.56
Emu3.5	8.11	7.70	7.59
GSTAR-Pro-8B	7.85	7.45	7.45

savings, we further optimize the system-level inference pipeline. We measure per-image latency for GSTAR-Flash-1.7B generating at 1024×1024 and apply four cumulative optimizations (Figure 5). Starting from a Causal AR head baseline of 22.23 s, replacing it with our Semi-AR Head reduces latency to 14.93 s ($1.49\times$) by parallelizing intra-group token prediction. The remaining optimizations target the two main bottlenecks separately: the intra-group head and the LLM backbone. For the head, we first capture each global step as a single CUDA graph, eliminating kernel-dispatch overhead ($2.55\times$, 5.85 s). We then fuse chains of small elementwise and reduction kernels in the sampling and normalization paths into composite Triton kernels, reducing the per-block kernel count and eliminating redundant global-memory round trips ($1.74\times$, 3.37 s). Finally, integrating the model into SGLang (Zheng et al., 2024) for LLM backbone serving yields a further $1.35\times$ speedup (2.50 s). The four stages compound to an $8.9\times$ end-to-end speedup.

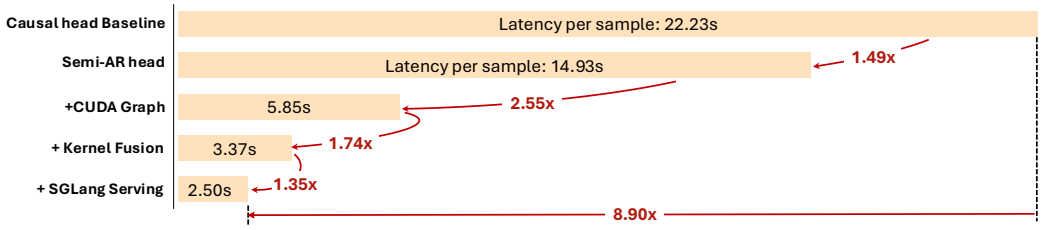


Figure 5: Cumulative speedup breakdown for GSTAR-Flash-1.7B at 1024×1024. Starting from the Causal AR head baseline (22.23 s), each stage—Semi-AR head, CUDA graph capture, fused top-k kernel, and SGLang serving—contributes multiplicative savings, reaching $8.9\times$ end-to-end.

6 Conclusion

We explore high-quality autoregressive image generation under deep spatial compression by scaling discrete token capacity. Combining grouped quantization with efficient semi-autoregressive intra-group modeling enables generation at $32\times$ and $64\times$ spatial downsampling. Our GSTAR-Pro-8B and GSTAR-Flash-1.7B models demonstrate the effectiveness of this approach in large-scale text-to-image generation, achieving performance competitive with leading models and a $20\times$ speedup over prior discrete autoregressive models.

AI USE STATEMENT

We used AI tools solely to polish the language of the manuscript, including grammar, clarity, and readability. All suggested edits were reviewed by the authors, who take full responsibility for the final content.

REPRODUCIBILITY STATEMENT

The main text describes our tokenizer design, intra-group modeling approach, and experimental results. Appendix A provides training configurations for the tokenizer and generation models, including the ImageNet ablation settings in Appendix A.3.

REFERENCES

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

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. URL https://arxiv.org/abs/2511.21631.

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

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. URL https://arxiv.org/abs/2301.00704.

Jian Chen, Yesheng Liang, and Zhijian Liu. Dflash: Block diffusion for flash speculative decoding. *arXiv preprint arXiv:2602.06036*, 2026a.

Junsong Chen, Jincheng Yu, Yitong Li, Shuchen Xue, Haozhe Liu, Jingyu Xin, Yuyang Zhao, Tian Ye, Zhangjie Wu, Zian Wang, Daquan Zhou, Ping Luo, Song Han, and Enze Xie. SANA-Video 2.0: Hybrid linear attention with attention residuals for efficient video generation. *arXiv preprint arXiv:2607.21553*, 2026b.

Junsong Chen, Yuyang Zhao, Jincheng Yu, Ruihang Chu, Junyu Chen, Shuai Yang, Xianbang Wang, Yicheng Pan, Daquan Zhou, Huan Ling, Haozhe Liu, Hongwei Yi, Hao Zhang, Muyang Li, Yukang Chen, Han Cai, Sanja Fidler, Ping Luo, Song Han, and Enze Xie. SANA-Video: Efficient video generation with block linear diffusion transformer. In *The Fourteenth International Conference on Learning Representations*, 2026c. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/41b93c59da0d0f835907fd661d419db2-Abstract-Conference.html.

Junyu Chen, Han Cai, Junsong Chen, Enze Xie, Shang Yang, Haotian Tang, Muyang Li, and Song Han. Deep compression autoencoder for efficient high-resolution diffusion models. In *International Conference on Learning Representations*, 2025a.

Junyu Chen, Dongyun Zou, Wenkun He, Junsong Chen, Enze Xie, Song Han, and Han Cai. DC-AE 1.5: Accelerating diffusion model convergence with structured latent space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024. URL https://arxiv.org/abs/2403.20330.

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*, 2025c. URL https://arxiv.org/abs/2501.17811.

Xin Cheng, Xingkai Yu, Chenze Shao, Jiashi Li, Yunfan Xiong, Yi Qian, Jiaqi Zhu, Shirong Ma, Xiaokang Zhang, Jiasheng Ye, Qinyu Chen, Chengqi Deng, Jiping Yu, Damai Dai, Zhengyan Zhang, Yixuan Wei, Yixuan Tan, Wenkai Yang, Runxin Xu, Yu Wu, Zhean Xu, Xuanyu Wang, Muyang Chen, Rui Tian, Xiao Bi, Zhewen Hao, Shaoyuan Chen, Huanqi Cao, Wentao Zhang, Anyi Xu, Huishuai Zhang, Dongyan Zhao, and Wenfeng Liang. Dspark: Confidence-scheduled speculative decoding with semi-autoregressive generation. *arXiv preprint arXiv:2607.05147*, 2026. URL https://arxiv.org/abs/2607.05147.

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

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*, 2025.

Sinan Du, Jiahao Guo, Bo Li, Shuhao Cui, Zhengzhuo Xu, Yifu Luo, Yongxian Wei, Kun Gai, Xinggang Wang, Kai Wu, and Chun Yuan. Vqrae: Representation quantization autoencoders for multimodal understanding, generation and reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30322–30334, June 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Du_VQRAE_Representation_Quantization_Autoencoders_for_Multimodal_Understanding_Generation_and_Reconstruction_CVPR_2026_paper.html.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

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 *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.09680.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. doi: 10.48550/arXiv.1508.06576.

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. URL https://arxiv.org/abs/2507.22058.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/a3bf71c7c63f0c3bcb7ff67c67b1e7b1-Abstract-Datasets_and_Benchmarks.html.

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity∞: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15733–15744, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/

html/Han_Infinity_Scaling_Bitwise_AutoRegressive_Modeling_for_High-Resolution_Image_Synthesis_CVPR_2025_paper.html.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

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. URL https://arxiv.org/abs/2403.05135.

Ideogram AI. Ideogram 4.0 Technical Details: Open model at the forefront of design. Official technical blog, June 2026. URL https://ideogram.ai/blog/ideogram-4.0/.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011. doi: 10.1109/TPAMI.2010.57.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision*, 2016.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Niantong Li, Guangzheng Hu, Weixu Qiao, Ying Ba, Qichen Hong, Shijun Shen, Jinlin Wang, Fan Zhou, Jianye Kang, Xin Shang, Ziyi He, Wei Wang, Dalin Li, Jiahao Li, Jie Zhang, Kaiyuan Gao, Kun Yan, Lihan Jiang, Ningyuan Tang, Shengming Yin, Tianhe Wu, Xiao Xu, Xiaoyue Chen, Yuxiang Chen, Yan Shu, Yanran Zhang, Yilei Chen, Yixian Xu, Zekai Zhang, Zhendong Wang, Zihao Liu, Zikai Zhou, Hongzhu Shi, Yi Wang, Bing Zhao, Hu Wei, Lin Qu, and Chenfei Wu. Qwen-Image-Bench: From generation to creation in text-to-image evaluation. *arXiv preprint arXiv:2605.28091*, 2026a. URL https://arxiv.org/abs/2605.28091.

Shufan Li, Greg Heinrich, Hanrong Ye, Yonggan Fu, Aditya Grover, Jan Kautz, and Pavlo Molchanov. Nemotron-labs-diffusion-image: Advancing masked discrete diffusion for high-resolution image synthesis. *arXiv preprint arXiv:2606.29814*, 2026b. URL https://arxiv.org/abs/2606.29814.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, Guopeng Li, Yuang Peng, Quan Sun, Jingwei Wu, Yan Cai, Zheng Ge, Ranchen Ming, Lei Xia, Xianfang Zeng, Yibo Zhu, Binxing Jiao, Xiangyu Zhang, Gang Yu, and Daxin Jiang. Step1X-Edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025. URL https://arxiv.org/abs/2504.17761.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: on the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67:220102, 2024. doi: 10.1007/s11432-024-4235-6. URL https://link.springer.com/article/10.1007/s11432-024-4235-6.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/663bce02a0050c4a11f1eb8a7f1429d3-Abstract-Conference.html.

Zhuoyan Luo, Fengyuan Shi, Yixiao Ge, Yujiu Yang, Limin Wang, and Ying Shan. Open-magvit2: An open-source project toward democratizing auto-regressive visual generation. *arXiv preprint arXiv:2409.04410*, 2024. URL https://arxiv.org/abs/2409.04410.

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 *Advances in Neural Information Processing Systems*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/bbe04d329d531e8814f1199098bd8fb6-Abstract-Conference.html.

Tianren Ma, Lin Long, Chuyan Chen, Mu Zhang, Junbo Zhao, Tong Zhang, and Qixiang Ye. drae: Representation autoencoder with hyper-spherical codes. *arXiv preprint arXiv:2607.22148*, 2026.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, 2022. doi: 10.18653/v1/2022.findings-acl.177. URL https://aclanthology.org/2022.findings-acl.177/.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *Advances in Neural Information Processing Systems*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/48b383b24230e0e6e649d9c98dae4d8c-Abstract-Conference.html.

Ziqi Pang, Tianyuan Zhang, Fujun Luan, Yunze Man, Hao Tan, Kai Zhang, William T. Freeman, and Yu-Xiong Wang. RandAR: Decoder-only autoregressive visual generation in random orders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 45–55, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Pang_RandAR_Decoder-only_Autoregressive_Visual_Generation_in_Random_Orders_CVPR_2025_paper.html.

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. URL https://arxiv.org/abs/2606.18249.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Fengyuan Shi, Zhuoyan Luo, Yixiao Ge, Yujiu Yang, Ying Shan, and Limin Wang. Scalable image tokenization with index backpropagation quantization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16037–16046, October 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Shi_Scalable_Image_Tokenization_with_Index_Backpropagation_Quantization_ICCV_2025_paper.html.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8317–8326, 2019.

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.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2404.02905.

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.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, 2017.

Jiangtao Wang, Zhen Qin, Yifan Zhang, Vincent Tao Hu, Björn Ommer, Rania Briq, and Stefan Kesselheim. Scaling image tokenizers with grouped spherical quantization. *arXiv preprint arXiv:2412.02632*, 2024.

Yuqing Wang, Shuhuai Ren, Zhijie Lin, Yujin Han, Haoyuan Guo, Zhenheng Yang, Difan Zou, Jiashi Feng, and Xihui Liu. Parallelized autoregressive visual generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

xAI. Grok-1.5 Vision Preview. https://x.ai/news/grok-1.5v, 2024. Introducing the RealWorldQA benchmark.

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. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2408.12528.

Zhihao Xie, Junfeng Wu, Xinting Hu, Junchao Huang, and Li Jiang. VideoRAE: Taming video foundation models for generative modeling via representation autoencoders. *arXiv preprint arXiv:2607.14088*, 2026.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15703–15712, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yao_Reconstruction_vs._Generation_Taming_Optimization_Dilemma_in_Latent_Diffusion_Models_CVPR_2025_paper.html.

Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David Ross, and Lu Jiang. Language model beats diffusion - tokenizer is key to visual generation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/036912a83bdbb1fd792baf6532f102d8-Abstract-Conference.html.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2411.00776.

Qihang Yu, Qihao Liu, Ju He, Xinyang Zhang, Yang Liu, Liang-Chieh Chen, and Xi Chen. Autoregressive image generation with masked bit modeling. *arXiv preprint arXiv:2602.09024*, 2026.

Z.ai Team. GLM-Image: Auto-regressive for dense-knowledge and high-fidelity image generation. Official model repository, 2026. URL https://github.com/zai-org/GLM-Image.

Huichao Zhang, Liao Qu, Yiheng Liu, Hang Chen, Yangyang Song, Yongsheng Dong, Shikun Sun, Xian Li, Xu Wang, Yi Jiang, Hu Ye, Bo Chen, Yiming Gao, Peng Liu, Akide Liu, Zhipeng Yang, Qili Deng, Linjie Xing, Jiyang Liu, Zhao Wang, Yang Zhou, Mingcong Liu, Yi Zhang, Qian He, Xiwei Hu, Zhongqi Qi, Jie Shao, Zhiye Fu, Shuai Wang, Fangmin Chen, Xuezhi Chai, Zhihua Wu, Yitong Wang, Zehuan Yuan, Daniel K. Du, and Xinglong Wu. Nextflow: Unified sequential modeling activates multimodal understanding and generation. *arXiv preprint arXiv:2601.02204*, 2026. URL https://arxiv.org/abs/2601.02204.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0u1LigJaab.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, 2024.

Shaobin Zhuang, Yiwei Guo, Fangyikang Wang, Canmiao Fu, Zhipeng Huang, Zeyue Tian, Xiaohui Li, Ying Zhang, Chen Li, and Yali Wang. Wetok: Powerful discrete tokenization for high-fidelity visual reconstruction. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0b2a58f2b5943b9c356f10db32073275-Abstract-Conference.html.

A Training Details

A.1 Model Training

Initialization and optimization. We initialize GSTAR-Pro-8B from Qwen3-8B and GSTAR-Flash-1.7B from Qwen3-1.7B. Both models share the same three-stage training schedule, consisting of pretraining (PT), continued training (CT), and supervised fine-tuning (SFT). We use ZeRO Stage 2 throughout all three stages and optimize both models using Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.95$.

Training schedule. We use any-resolution training throughout all three stages, progressively increasing the image resolution. PT runs for 400k steps at 512px, followed by 100k CT steps at 1024px. Both stages use a sequence length of 4,096, a global batch size of 64, and a learning rate of 10^{-4}. SFT runs for 30k steps, mixing 1024px and 2048px images at a ratio of 1:1, with a sequence length of 8,192, a global batch size of 32, and a learning rate of 10^{-5}. During SFT, image-editing and text-to-image generation data are mixed at a ratio of 3:7. Table 6 summarizes the configurations shared by both models.

Table 6: Training configurations shared by GSTAR-Pro-8B and GSTAR-Flash-1.7B. All stages use any-resolution training. SFT mixes the 1024px and 2048px resolution tiers at a ratio of 1:1.

Stage	Resolution	Steps	Seq. length	Global batch	LR
PT	512px	400k	4,096	64	10^{-4}
CT	1024px	100k	4,096	64	10^{-4}
SFT	1024px / 2048px	30k	8,192	32	10^{-5}

Training data. For PT and CT, we collect image–text pairs from publicly available datasets, including LAION, DataComp, and JourneyDB, and regenerate their captions using Qwen3.5-122B-A10B. For SFT, we select a higher-quality subset of these data and supplement it with a small proportion of images regenerated using Seedream and Nano Banana 2.

A.2 Tokenizer Training

Initialization and objectives. We initialize the tokenizer encoder from Qwen-ViT with a spatial downsampling factor of 32. For both the ImageNet ablations and text-to-image tokenizers, we use the same loss weights: $\lambda_{\text{sem}} = 1$, $\lambda_{\text{gram}} = 10$, $\lambda_{\text{perceptual}} = 1$, $\lambda_{\text{vq}} = 1$, and $\lambda_{\text{commit}} = 0.25$. Here $\lambda_{\text{perceptual}}$ weights the perceptual reconstruction loss, and λ_{commit} weights the commitment term within the vector-quantization objective.

Optimization. All tokenizers are trained with Adam using $(\beta_1, \beta_2) = (0.9, 0.95)$ and a global batch size of 1,024. We use a base learning rate of 10^{-4} for the quantizer, decoder, and other components. To preserve the pretrained semantic encoder's semantic representations, we use a lower learning rate of 10^{-5} for the encoder.

Tokenizer configurations. For GSTAR-Pro-8B, we train a $32\times$ tokenizer with 16 quantization groups per spatial position. For GSTAR-Flash-1.7B, we train a $32\times$ tokenizer with 8 quantization groups per spatial position, then merge each 2×2 neighborhood of quantized positions to obtain an effective $64\times$ tokenizer with 32 groups per global position. This merging operation reorganizes the quantized representation without introducing an additional quantization step.

Tokenizers for text-to-image generation. We use any-resolution training for the text-to-image tokenizers. In addition to general-purpose image data, we incorporate OCR, aesthetic, and face data to improve generalization and text-rendering capabilities. Training runs for a total of 200k steps.

Following UniAR (Peng et al., 2026) and Emu3.5 (Cui et al., 2025), we freeze the tokenizer after training and train a DiT-based visual decoder initialized from Ideogram 4 (Ideogram AI, 2026), conditioned on the discrete tokens produced by the frozen tokenizer. This decoder is used solely to refine visual quality. Unlike GLM-Image (Z.ai Team, 2026) and X-Omni (Geng et al., 2025), our decoder conditions exclusively on our discrete visual tokens, without any additional textual conditioning. The ImageNet ablation settings are described separately in Appendix A.3.

A.3 ImageNet Ablation Settings

Tokenizer training. For the ImageNet ablations, we train the tokenizers on ImageNet for 100k steps at a fixed resolution of 256×256. The loss weights, optimizer, component-wise learning rates, and global batch size follow Appendix A.2.

Class-conditional generation. For the class-to-image (C2I) ablations, we train the autoregressive generator on ImageNet at 256×256 resolution using the Llama-XL architecture from LlamaGen (Sun et al., 2024). We train for 50k steps with a sequence length of 4,096, a global batch size of 128, and a learning rate of 4×10^{-4}. All remaining hyperparameters follow the Llama-XL configuration.

B Multimodal Understanding

Experimental setup. We initialize the language model from Qwen3-1.7B and train on the standard LLaVA-OneVision training data (Li et al., 2024). We compare continuous visual representations from the original Qwen3 ViT (without further training) with discrete representations produced by our f32g16 tokenizer, which uses a spatial downsampling factor of 32 and 16 quantization groups per position.

Results. As shown in Table 7, our discrete tokenizer achieves multimodal understanding performance comparable to continuous visual representations, with average scores of 64.7 and 64.6, respectively. These results suggest that our tokenizer largely preserves the semantic information needed for multimodal understanding despite discretization and spatial compression.

Table 7: Multimodal understanding performance using continuous and discrete visual representations. Higher is better for all metrics. The average is computed over the eight listed benchmarks, with OCRBench divided by 10 to match the 0–100 scale of the other scores.

Benchmark	Continuous	Discrete (f32g16)
MathVista (Lu et al., 2024)	45.7	44.5
RealWorldQA (xAI, 2024)	60.1	59.3
MMStar (Chen et al., 2024)	45.8	44.7
OCRBench (Liu et al., 2024)	676	665
TextVQA (Singh et al., 2019)	71.3	71.0
AI2D (NO_MASK) (Kembhavi et al., 2016)	76.7	78.6
ChartQA (Masry et al., 2022)	67.4	69.9
DocVQA (Mathew et al., 2021)	81.9	82.9
Average	64.6	64.7

C Evaluation Protocols and Additional Results

C.1 Latency Measurement

We measure batch-one generation at 1024×1024 on an NVIDIA H200 and separately report global-backbone time, total intra-group head time, image-decoder time, and end-to-end pipeline latency.

Table 8: Text-to-image generation with lossless latent packing. The configuration with $64\times$ spatial compression and $G = 32$ is obtained by losslessly packing the configuration with $32\times$ spatial compression and $G = 8$, leaving tokenizer reconstruction unchanged. PRISM is the combined score on a 0–10 scale.

Compression	G	GenEval ↑	DPG ↑	PRISM ↑
$32\times$	8	0.8060	82.49	5.30
$64\times$	32	0.7885	80.44	5.13

All models run in BF16 precision. The software stack consists of PyTorch 2.9.1 with CUDA 12.8, Triton 3.5.1, and Transformers 4.57.6. The decode loop is captured as a single CUDA graph. Each configuration is warmed up for 3 iterations, then timed over 3 independent trials; we report the median latency. Sampling uses classifier-free guidance with a scale of 1.8, temperature 0.5, and top-k = 1024. We report run-to-run variation with the final measurements. Matched comparisons use identical timing boundaries and implementation settings.

D Lossless Packing at $64\times$ Spatial Compression

We evaluate whether our lossless packing strategy can increase the spatial compression factor while preserving both tokenizer reconstruction and text-to-image generation performance. Starting from $\mathrm{f32g8}$ (i.e., $32\times$ spatial compression with $G = 8$; other configurations follow the same notation), we pack each 2×2 neighborhood of spatial latent positions along the group dimension to obtain $\mathrm{f64g32}$. Specifically, four spatial positions with eight groups each are rearranged into a single spatial position with 32 groups. This operation is exactly invertible and neither discards nor re-quantizes any latent codes. Consequently, applying the inverse packing operation before decoding yields identical tokenizer reconstruction by construction, while reducing the number of spatial latent positions by $4\times$.

To assess the effect of packing on generation, we train the text-to-image generator with the $\mathrm{f64g32}$ representation using the same training recipe as for the $\mathrm{f32g8}$ baseline. The results are reported in Table 8.

As shown in Table 8, the packed $\mathrm{f64g32}$ representation achieves a GenEval score of 0.7885, a DPG score of 80.44, and a PRISM score of 5.13. Compared with $\mathrm{f32g8}$, it retains strong generative performance despite using a $4\times$ shorter spatial latent sequence. These results demonstrate that our lossless packing strategy enables effective text-to-image generation at a $64\times$ spatial compression factor without altering the reconstruction capability of the tokenizer.

E IMAGE RECONSTRUCTION QUALITY

We also evaluate the reconstruction capability of our visual tokenizers on the ImageNet validation set at a fixed resolution of 512×512. We report peak signal-to-noise ratio (PSNR) and compare against five open-source visual tokenizers: VQGAN (Esser et al., 2021), IBQ (Shi et al., 2025), VAR (Tian et al., 2024), Open-MAGVIT2 (Luo et al., 2024), and Emu3.5 (Cui et al., 2025).

As shown in Table 9, our `f32g16` tokenizer achieves the highest PSNR of 25.12 dB at $32\times$ effective spatial compression, exceeding Open-MAGVIT2 by 1.14 dB. At $64\times$ effective spatial compression, our `f64g32` variant achieves 23.90 dB, outperforming VQGAN, IBQ, VAR, and Emu3.5 while remaining within 0.08 dB of Open-MAGVIT2. These results show that our tokenizers retain competitive reconstruction fidelity at substantially higher effective spatial compression factors.

Table 9: Image reconstruction quality on the ImageNet validation set at 512×512 resolution. We report the effective per-axis spatial compression factor; for VAR it refers to the finest latent scale. Higher PSNR is better.

Tokenizer	Spatial compression	PSNR (dB) ↑
VQGAN (Esser et al., 2021)	16×	22.24
IBQ (Shi et al., 2025)	16×	23.39
VAR (Tian et al., 2024)	16×	23.43
Open-MAGVIT2 (Luo et al., 2024)	16×	23.98
Emu3.5 (Cui et al., 2025)	16×	23.82
GSTAR-Tokenizer (Ours) `f32g16`	32×	**25.12**
GSTAR-Tokenizer (Ours) `f64g32`	64×	23.90

We also present qualitative reconstruction examples in Figure 6, comparing the ViT decoder and the refiner for both the `f32g16` and `f64g32` tokenizers. The examples illustrate how the refiner improves visual detail and sharpness while preserving the overall image content.



Figure 6: Qualitative reconstruction examples. From left to right: ground truth, `f32g16` ViT decoder, `f32g16` refiner, `f64g32` ViT decoder, and `f64g32` refiner. Each row shows the same example across all methods. These visual examples are shown at 1024×1024 resolution, separately from the quantitative ImageNet evaluation at 512×512.