BitDance: Scaling Autoregressive Generative Models with Binary Tokens

Yuang Ai[*2,3] **Jiaming Han**[*2,3] **Shaobin Zhuang**[*3,4] **Weijia Mao**[3,5]
Xuefeng Hu[3] **Ziyan Yang**[3] **Zhenheng Yang**[3] **Yali Wang**[7]
Hao Chen[*3†‡] **Xiangyu Yue**[2,6†] **Huaibo Huang**[1†]

[1]Institute of Automation, Chinese Academy of Sciences
[2]MMLab, The Chinese University of Hong Kong [3]ByteDance
[4]Shanghai Jiao Tong University
[5]National University of Singapore [6]Shenzhen Loop Area Institute
[7]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

[*]Equal contribution [†]Corresponding Author [‡]Project lead

Abstract

We present BitDance, a scalable autoregressive (AR) image generator that predicts binary visual tokens instead of codebook indices. With high-entropy binary latents, BitDance lets each token represent up to $\mathbf{2^{256}}$ states, yielding a compact yet highly expressive discrete representation. Sampling from such a huge token space is difficult with standard classification. To resolve this, BitDance uses a binary diffusion head: instead of predicting an index with softmax, it employs continuous-space diffusion to generate the binary tokens. Furthermore, we propose next-patch diffusion, a new decoding method that predicts multiple tokens in parallel with high accuracy, greatly speeding up inference. On ImageNet 256×256, BitDance achieves an FID of **1.24**, the best among AR models. With next-patch diffusion, BitDance beats state-of-the-art parallel AR models that use 1.4B parameters, while using **5.4**× fewer parameters (260M) and achieving **8.7**× speedup. For text-to-image generation and image editing, BitDance trains on large-scale multimodal tokens and generates high-resolution, photorealistic images efficiently, showing strong performance and favorable scaling. When generating 1024×1024 images, BitDance achieves a speedup of over **30**× compared to prior AR models. We release code, models, and demo at https://bitdance.csuhan.com.

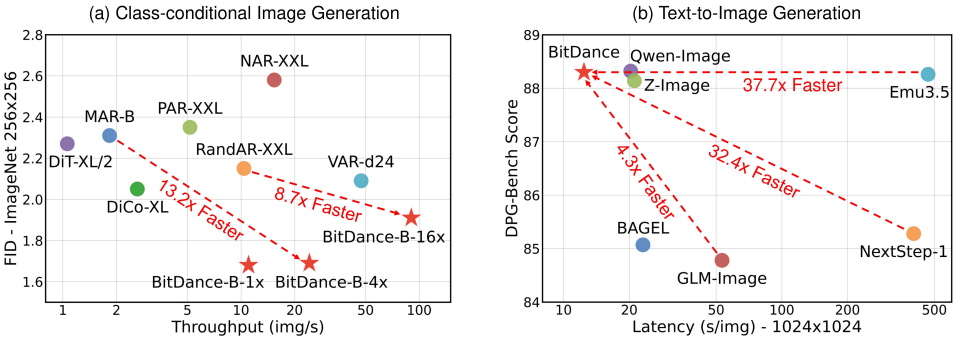


Figure 1: **Performance vs. efficiency** compared with SOTA diffusion models and autoregressive models.

1 Introduction

Large language models (LLMs) have shown that autoregressive pre-training via next-token prediction can distill large-scale textual knowledge into a single model with strong generalization [1, 15, 29].

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



Figure 2: **Showcase of BitDance** in image generation, complex image editing, and free-form manipulation.

This success has motivated extending AR modeling to visual generation [64, 69, 45], yielding a unified paradigm with language modeling and enabling progress on multimodal applications [22, 66, 89].

However, applying the AR paradigm to vision still faces persistent challenges. A primary issue is token design: vision tokens must be expressive enough to capture rich image content while remaining sufficiently regularized to limit error accumulation over long sequences. Existing discrete AR models typically leverage Vector Quantization (VQ) [70, 21] for tokenization, but often suffer from degraded reconstruction due to difficulties in scaling visual vocabularies [64, 69, 71]. In contrast, while continuous AR models use VAEs to achieve superior reconstruction, the unconstrained nature of their latent spaces often leads to severe error accumulation [58, 38, 66]. Furthermore, the sequential nature of token-by-token generation imposes a significant bottleneck on inference efficiency, particularly as image resolutions scale upward.

In this paper, we present BitDance, a simple yet scalable autoregressive framework that achieves state-of-the-art performance in image generation. BitDance is built upon three key components: **(i)** a large-vocabulary binary tokenizer, **(ii)** a binary diffusion head for sampling in extremely large discrete spaces, and **(iii)** a next-patch diffusion paradigm that enables efficient multi-token prediction.

Inspired by recent advances in binary quantization [85, 91], we scale up the entropy of binary representations, expanding the vocabulary size up to 2^{256}. This scale is orders of magnitude larger than that of existing discrete tokenizers [64, 31, 72], enabling our discrete representation to surpass continuous VAEs in reconstruction fidelity (see Tab. 1). By encoding images into a compact yet expressive binary latent space, BitDance preserves fine-grained visual details while introducing beneficial discreteness that helps regularize long-sequence generation and mitigate error accumulation.

However, such an expansive vocabulary poses severe challenges for sampling. As shown in Fig. 3, index-based classification heads suffer from an inherent trade-off between parameter efficiency and sampling accuracy. To address this issue, we propose a binary diffusion head. Rather than mapping binary tokens to discrete indices, we embed binary tokens as vertices of a hypercube in continuous space and model their distribution using a diffusion objective [34]. By jointly modeling all binary channels, our head enables accurate sampling while maintaining a controllable parameter footprint.

To further accelerate vision token prediction, we propose a next-patch diffusion approach for visual autoregressive modeling. Considering the spatial dependency in images, we posit that tokens within a local patch are highly correlated, making them easier to predict jointly. As shown in Fig. 5, unlike prior parallel AR generation methods [73, 55, 43] that use factorized sampling—sampling each token independently via a classification head—BitDance uses a binary diffusion head to explicitly model the joint distribution of tokens generated in parallel. This design enables high-fidelity parallel token prediction and substantially improves efficiency.

BitDance demonstrates strong performance across diverse generative tasks. On ImageNet 256×256, the 1B model achieves an FID of 1.24, surpassing previous AR models. As shown in Fig. 1, BitDance outperforms a 1.4B state-of-the-art parallel AR model with only 260M parameters, while being 8.7× faster. For text-to-image generation and image editing, we scale BitDance to 14B parameters and train it on large-scale multimodal tokens. After multi-stage training, it shows strong high-resolution image synthesis capability with fine-grained details (see Fig. 2). BitDance achieves 0.86 on GenEval [27], 88.28 on DPG-Bench [36], 0.532 on OneIG-EN [8], and 0.512 on OneIG-ZH [8], establishing a new state-of-the-art among autoregressive models. As further shown in Fig. 1, it delivers over 30× faster generation at 1024×1024 than standard next-token AR models.

Our main contributions can be summarized as follows:

- We present BitDance, a simple and scalable autoregressive model. BitDance demonstrates the viability of scaling token entropy for high-fidelity visual generation, offering new insights into the design space of visual autoregressive modeling.
- We propose a binary diffusion head to resolve the sampling bottlenecks inherent in expansive visual vocabularies. By seamlessly extending it to multi-token sampling, BitDance facilitates precise and efficient parallel prediction through a next-patch diffusion paradigm.
- Extensive experiments on diverse generative tasks show that BitDance offers exceptional scaling, achieving superior generative quality with high inference speed.

2 Related Work

Autoregressive Visual Generation. Standard autoregressive (AR) visual generation typically quantizes images into discrete tokens and models their distribution via next-token prediction in a raster-scan order [64, 71, 16]. Recently, several paradigm shifts have emerged, exploring continuous token spaces [45, 59], randomized ordering [87, 55, 43], and alternative modeling primitives [69, 58]. For instance, MAR [45] introduces a token-level diffusion head to facilitate continuous token sampling, while NextStep-1 [66] scales up the continuous AR framework for high-fidelity text-to-image synthesis. However, continuous tokens often lack sufficient regularization, leading to severe error accumulation and representation drift [58, 65] during long-sequence generation—factors that significantly degrade the quality of high-resolution images. SphereAR [38] employs hyperspherical constraints to regularize the latent features of VAEs. In this work, we focus on discrete binary representations to explore the potential of this more radical constraint in AR visual generation.

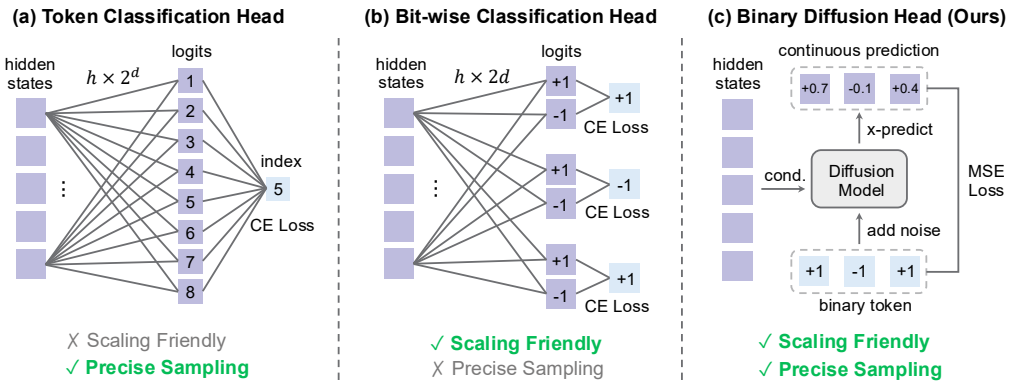


Figure 3: **Binary token sampling paradigms.** Scaling up binary token entropy yields superior reconstruction performance, but it simultaneously creates a bottleneck during sampling. For a d-channel binary token: (a) Directly modeling $p(b_1, b_2, \ldots, b_d)$ requires $h \times 2^d$ parameters, which suffers from an exponential explosion as d scales. (b) Bit-wise classification [31] reduces the parameter count to $h \times 2d$ by assuming bit independence, i.e., $\prod_{i=1}^{d} p(b_i)$, but this restrictive assumption compromises sampling fidelity. (c) We embed binary tokens as vertices of a d-dimensional hypercube in continuous space. By modeling the joint distribution of all bits via a diffusion objective, we achieve controllable parameters as d scales up and high-fidelity sampling.

Parallel Prediction in AR Models. Accelerating AR generation has become a pivotal research direction in visual generation. Mask-GIT [7] and MAR [45] adopt MAE-style masking strategies for modeling. VAR [69] utilizes next-scale prediction to predict tokens within a unified scale in parallel. PAR [73] employs a grouping strategy to generate weakly dependent tokens. RandAR [55] and ARPG [43] leverage random-order modeling to enable the prediction of tokens at arbitrary positions. While showing promise, these methods often struggle to model the joint distribution of tokens predicted in parallel, lacking sufficient multi-token constraints during the final sampling stage (see Fig. 5). Our BitDance seamlessly extends the binary diffusion head to model the joint distribution of multiple tokens, achieving efficient and reliable parallel prediction through next-patch diffusion.

3 BitDance Architecture

3.1 Binary Visual Tokenizer

For discrete visual tokenizers, scaling up the vocabulary size to increase token entropy is critical for enhancing both reconstruction fidelity and downstream generation quality. However, traditional Vector Quantization (VQ) often encounters codebook collapse as the vocabulary expands. To ensure efficient utilization of the codebook space, we adopt binary quantization via Lookup-Free Quantization (LFQ) [85]. Given an encoded latent token $x \in \mathbb{R}^d$, LFQ employs an implicit, learning-free codebook $\mathcal{C}_{LFQ} = \{-1, 1\}^d$. The quantization process is performed independently across each channel of x:

$$x_q = \text{sign}(x). \tag{1}$$

To prevent codebook collapse and maximize information capacity, entropy loss is commonly used:

$$\mathcal{L}_{entropy} = \mathbb{E}[H(q(x))] - H[\mathbb{E}(q(x))], \tag{2}$$

where $H(\cdot)$ denotes entropy calculation. In LFQ, computing the distribution $q(x)$ requires calculating similarities between x and the entire codebook space. As the vocabulary size grows exponentially (2^d), this calculation becomes computationally prohibitive due to its significant memory footprint.

To address this bottleneck, we adopt a group-wise LFQ strategy [94], which partitions d channels into g distinct groups for entropy calculation. This strategy strikes a balance between computational efficiency and optimization accuracy, enabling the training of models with an exceptionally large vocabulary. Specifically, we scale the codebook size up to $\mathbf{2^{256}}$. The vast representation space allows our discrete tokenizer to achieve reconstruction fidelity comparable to continuous VAEs (see Tab. 1).

3.2 Binary Diffusion Head

While a large vocabulary increases token entropy and reconstruction fidelity, it introduces challenges for sampling. For a binary token with d channels, the number of possible discrete indices grows

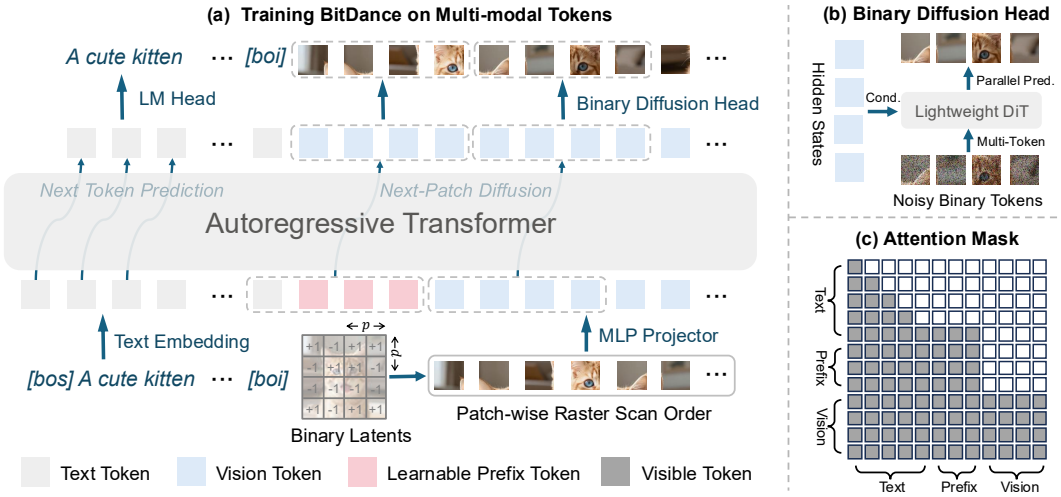


Figure 4: **Architecture of BitDance,** an autoregressive model trained on multi-modal tokens. An input image is first encoded into binary latents and then flattened into a 1D sequence following a patch-wise raster scan order with patch size $p \times p$. Vision tokens are modeled using our proposed next-patch diffusion, utilizing a binary diffusion head to achieve efficient and precise parallel prediction.

exponentially to 2^d. As illustrated in Fig. 3, conventional index-based classifiers face an inherent trade-off between sampling precision and parameter efficiency.

Sampling Bottleneck. To model the joint probability $p(b_1, b_2, \ldots, b_d)$ of a d-bit token, a standard classification head must characterize a categorical distribution across all 2^d possible indices. This approach incurs a prohibitive parameter overhead of $h \times 2^d$, where h denotes the hidden dimension. As d scales, this becomes computationally intractable; for instance, with $h = 1024$ and $d = 32$, the parameter count reaches approximately 4.4 trillion, far exceeding the capacity of modern hardware.

To address this, [31] adopts a bit-wise independence assumption, decomposing the joint distribution into d separate binary classifications. By modeling the distribution as a product of independent probabilities, $\prod_{i=1}^{d} p(b_i)$, the approach effectively reduces the parameter overhead to $h \times 2d$. However, this assumption is overly restrictive, as it fails to capture the intricate inter-bit correlations in binary tokens, leading to degraded sampling precision and inferior generative quality (see Tab. 7).

Diffusion-based Binary Prediction. We propose a novel binary diffusion head to address these sampling difficulties. Unlike conventional applications of diffusion models that primarily focus on modeling continuous data distributions, we are the first to leverage such models for discrete binary tokens. Rather than mapping these tokens to discrete indices, we represent them as d-dimensional vectors within a continuous space. Specifically, let $x \in \mathbb{R}^d$ denote the ground-truth binary token to be predicted and $z \in \mathbb{R}^h$ represent the condition (i.e., the hidden state from the AR transformer). To model the conditional probability distribution $p(x|z)$, we adopt the Rectified Flow [48] formulation and optimize the head using x-prediction with a velocity-matching loss [44]:

$$\mathcal{L}(z, x) = \mathbb{E}_{t,x,\epsilon} \left\| v_\theta(x_t, t, z) - v_t \right\|^2, \tag{3}$$

where $x_t = tx + (1-t)\epsilon$ is the noisy token at time t interpolated with Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, and $v_t = x - \epsilon$ is the target velocity. The velocity v_θ is parameterized by a x-prediction network f_θ such that $v_\theta(x_t, t, z) = (f_\theta(x_t, t, z) - x_t)/(1-t)$.

During inference, we initialize $x_0 \sim \mathcal{N}(0, \mathbf{I})$ and integrate the learned velocity field using an Euler solver with N uniform steps $\Delta_t = 1/N$:

$$x_{t+\Delta_t} = x_t + v_\theta(x_t, t, z)\Delta_t. \tag{4}$$

After N steps, we apply a **hard binarization constraint**: $x_1 = \text{sign}(x_1)$. This step effectively projects the continuous prediction back onto the binary hypercube, exploiting the structural priors of the token space. This mechanism circumvents the compounded error accumulation typically found in AR models that operate in unconstrained continuous spaces [38, 66].

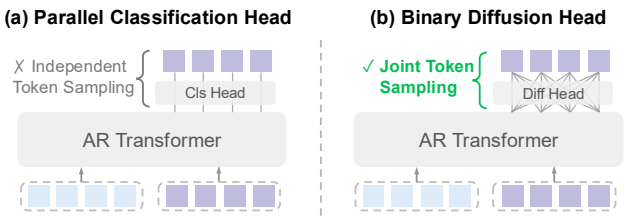


Figure 5: **Comparison of sampling heads for parallel prediction.** (a) Standard classification head is limited to independent token sampling, which violates the inherent dependencies required for parallel prediction. (b) Our proposed binary diffusion head models the joint distribution of tokens generated simultaneously, enabling coherent sampling.

Why Binary Diffusion Head is Effective. Geometrically, binary tokens in continuous space form a finite set of vertices on a hypercube, characterized by uniform magnitudes but distinct orientations. Unlike standard VAE latent spaces which are often unconstrained, the binary target space provides a strong structural regularization. This restriction to a finite, well-defined set of target vectors significantly reduces the optimization complexity, leading to faster convergence and superior sampling stability (see Tab. 6). Fig. 7 illustrates the output distribution of the head. It reveals that the model effectively learns the discrete nature of binary tokens without any manual constraints.

3.3 Next-Patch Diffusion

Standard autoregressive (AR) generation formulates image synthesis as a sequence of next-token predictions. Formally, given a flattened 1D sequence of N tokens $x = [x_1, x_2, \ldots, x_N]$, the generative process is decomposed into a chain of conditional probabilities:

$$p(x) = \prod_{n=1}^{N} p(x_n | x_1, x_2, \ldots, x_{n-1}). \tag{5}$$

While effective, this token-by-token paradigm imposes a significant inference bottleneck, particularly as image resolutions and sequence lengths increase.

Next-Token to Next-Patch. To alleviate this, we exploit the inherent spatial dependency in visual data. We posit that tokens within a local patch exhibit strong statistical dependencies and can thus be predicted in parallel without compromising generative quality. In our next-patch prediction scheme, the sequence x is partitioned into M disjoint groups (patches), $x = [X_1, X_2, \ldots, X_M]$, where each group $X_m = \{x_m^1, x_m^2, \ldots, x_m^{p^2}\}$ contains $p \times p$ tokens. The parallel generation can be expressed as:

$$p(x) = \prod_{m=1}^{M} p(X_m \mid X_1, \ldots, X_{m-1}). \tag{6}$$

As shown in Fig. 4, we adopt a patch-wise raster-scan order and realize it in the AR Transformer using a block-wise causal attention mask. Unlike the standard causal mask with strict 1D dependencies, ours allows mutual visibility within each patch, capturing intra-patch spatial interactions while preserving autoregressive dependencies across patches. We also prepend $p^2 - 1$ learnable prefix tokens to enable simultaneous prediction of all tokens in the first patch.

Precise Multi-token Sampling. Recent parallel AR frameworks [73, 55, 43] propose diverse modeling trajectories and specialized attention designs, but their performance remains fundamentally limited by a **training–inference discrepancy** in the parallel AR objective. They still optimize the token-wise objective in Eq. (5), whereas inference requires simultaneous generation of token groups. As illustrated in Fig. 5, tokens within each group are nevertheless **sampled independently** using a standard classification head. This implicit independence assumption conflicts with the **joint distribution modeling** required by Eq. (6), leading to reduced structural coherence and artifacts within the generated groups.

To bridge this gap, we extend the binary diffusion head introduced in Sec. 3.2 to support joint multi-token prediction. By leveraging the flexibility of the diffusion formulation, we adapt the training objective in Eq. (3) to multi-token. Let $X \in \mathbb{R}^{p^2 \times d}$ represent the patch of ground-truth tokens and $Z \in \mathbb{R}^{p^2 \times h}$ denote the corresponding hidden states. The optimization objective becomes:

$$\mathcal{L}_{parallel} = \mathbb{E}_{t,X,\epsilon} \, ||v_\theta(X_t, t, Z) - v_t| \, |^2. \tag{7}$$

To effectively model the p^2 tokens within the head, we design the architecture of prediction network f_θ as a lightweight DiT [56]. By aligning the training objective with the diffusion head's capacity for **joint distribution modeling**, BitDance achieves efficient and accurate parallel prediction.

4 Experiments

4.1 Scaling up Token Entropy

We study the effect of scaling token entropy in binary visual tokenizers under three settings: (i) 16× downsampling with a codebook size of $\mathbf{2^{32}}$, (ii) 32× downsampling with $\mathbf{2^{128}}$, and (iii) 32× downsampling with $\mathbf{2^{256}}$. We provide more training details of tokenizers in Appendix A.

Table 1: **Reconstruction on ImageNet 256×256 validation set.** All models are trained on general domain datasets. D: discrete, C: continuous, CB: codebook, Comp.: compression. By scaling token entropy, our discrete tokenizer outperforms continuous VAEs, even at higher compression ratios.

Method	Type	Down Ratio	CB Size	Comp. Ratio	PSNR↑	SSIM↑
SD-VAE [60]	C	8	-	24	23.54	0.68
Cosmos [2]	D	16	65536	384	19.93	0.245
Show-o [80]	D	16	8192	473	21.34	0.59
LlamaGen [64]	D	16	16384	439	20.65	0.54
Open-MAGVIT2 [50]	D	16	2^{18}	341	22.70	0.64
Infinity [31]	D	16	2^{32}	192	22.70	-
BitDance-Tok	D	16	$\mathbf{2^{32}}$	192	**24.90**	**0.72**
WeTok [94]	D	32	2^{32}	768	20.77	0.55
DC-AE [9]	C	32	-	48	24.81	0.69
DC-AE-SANA [78]	C	32	-	48	24.72	0.69
BitDance-Tok	D	32	$\mathbf{2^{128}}$	192	23.26	0.67
BitDance-Tok	D	32	$\mathbf{2^{256}}$	96	**25.29**	**0.74**

Scaling Token Entropy for Reconstruction. Tab. 1 compares tokenizers across different settings. Scaling token entropy significantly improves discrete reconstruction quality and bridges the gap to continuous models. Our 16× tokenizer with a 2^{32} codebook outperforms continuous SD-VAE [60], while under 32× downsampling, enlarging the codebook from 2^{128} to 2^{256} enables our discrete tokenizer to surpass continuous DC-AE [9].

Scaling Token Entropy and AR Model Size. We further examine the effect of vocabulary size on ImageNet generation. To align training FLOPs across downsampling rates, we replicate class tokens during training. Fig. 6 shows that small Transformers struggle with large vocabularies, whereas larger models benefit from them and achieve better generative quality. Thus, vocabulary scaling is most effective when accompanied by larger Transformer scale.

4.2 Class-conditional Image Generation

We experiment on ImageNet at 256×256 resolution. We consider three model variants, BitDance-B/L/H, with parameter counts roughly matched to their MAR [45] counterparts. Our model names also specify the patch size p used in next-patch diffusion. For example, BitDance-B-4x denotes $p = 2$, where each step generates $p^2 = 4$ tokens. More details are provided in Appendix B.

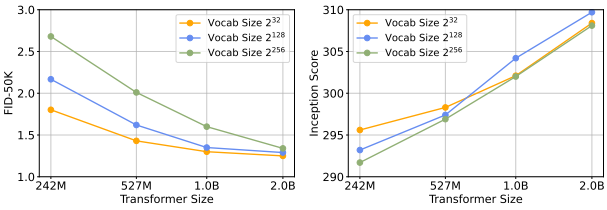


Figure 6: **Generative performance across different vocabulary sizes and Transformer sizes.** Scaling the vocabulary size requires a concurrent expansion of the Transformer size.

We first examine scaling with $p = 1$. Tab. 2 shows consistent gains with model size, and our largest model, BitDance-H-1x (1B), achieves an FID of 1.24, outperforming prior AR baselines under the standard raster-scan causal setting. For parallel generation, BitDance-B-4x outperforms RandAR-XXL [55] by about 0.5 FID with only 260M parameters versus 1.4B (Tab. 3), demonstrating effective joint modeling of parallel tokens by the binary diffusion head.

4.3 Text-to-Image Generation and Editing

To further validate BitDance, we scale the AR model to 14B Qwen3 [82], using the pretrained LLM as the text encoder and extending it with image generation capability. We set $p = 4$, allowing parallel prediction of 16 tokens per step. Training details are given in Appendix C.

Table 2: **Evaluation of class-conditional image generation on ImageNet 256×256.** We mark models using additional self-supervised models (e.g., DINO) in gray. BitDance achieves superior performance while employing standard AR modeling with raster-order. Detailed results are provided in Tab. 9.

Model	Type	Order	Tokenizer	#Params	FID↓	IS↑	Pre.↑	Rec.↑
Continuous Tokens								
DiT-XL/2 [56]	Diff.	-	VAE	675M	2.27	278.2	0.83	0.57
REPA [88]	Diff.	-	VAE	675M	1.42	305.7	0.80	0.65
RAE [92]	Diff.	-	RAE	675M	1.13	262.6	0.78	0.67
MAR-H [45]	Mask.	random	VAE	943M	1.55	303.7	0.81	0.62
xAR-H [58]	AR+Diff.	raster	VAE	1.1B	1.24	301.6	0.83	0.64
SphereAR-H [38]	AR	raster	VAE	943M	1.34	300.0	0.80	0.64
Discrete Tokens								
LlamaGen-XXL [64]	AR	raster	VQ	1.4B	2.34	253.9	0.80	0.59
RAR-XXL [87]	AR	hybrid	VQ	1.5B	1.48	**326.0**	0.80	0.63
OpenMAGVIT2-XL [50]	AR	raster	LFQ	804M	2.51	271.7	**0.84**	0.54
MAGVIT-v2 [85]	Mask.	random	LFQ	307M	1.78	319.4	-	-
VAR-d30 [69]	VAR	-	VQ	2B	1.92	323.1	0.82	0.59
BitDance-B-1x	AR	raster	LFQ	242M	1.68	297.1	0.80	0.62
BitDance-L-1x	AR	raster	LFQ	527M	1.31	300.2	0.80	0.64
BitDance-H-1x	AR	raster	LFQ	1.0B	**1.24**	304.4	0.81	**0.64**

Table 3: **Overall comparison with parallel image generation methods on ImageNet 256×256.** Throughput is measured on one A100 with a batch size of 64 at bfloat16 precision. With only 260M parameters, our model outperforms the 1.4B SOTA model, while achieving faster generation speeds. Refer to Tab. 10 for full results.

Model	Type	Order	#Params	Steps	Throughput↑	FID↓	IS↑	Pre.↑	Rec.↑
DiT-XL/2 [56]	Diff.	-	675M	250	1.06 img/s	2.27	278.2	0.83	0.57
DiCo-XL [3]	Diff.	-	701M	250	2.62 img/s	2.05	282.2	0.83	0.59
MaskGIT [7]	Mask.	random	227M	8	50.73 img/s	6.18	182.1	0.80	0.51
MAR-L [45]	Mask.	random	479M	256	1.39 img/s	1.78	296.0	0.81	0.60
VAR-d20 [69]	VAR	-	600M	10	71.31 img/s	2.57	302.6	**0.83**	0.56
VAR-d24 [69]	VAR	-	1.0B	10	47.22 img/s	2.09	312.9	0.82	0.59
PAR-XXL [73]	AR	hybrid	1.4B	147	5.17 img/s	2.35	263.2	0.80	0.62
NAR-XXL [32]	AR	hybrid	1.5B	31	15.37 img/s	2.58	293.5	0.82	0.57
RandAR-XXL [55]	AR	random	1.4B	88	10.39 img/s	2.15	**322.0**	0.79	0.62
BitDance-B-4x	AR	raster	260M	64	24.18 img/s	**1.69**	291.2	0.79	**0.63**
BitDance-B-16x	AR	raster	260M	16	**90.26** img/s	1.91	283.8	0.78	0.62

Table 4: **Evaluation of text-to-image generation** on DPG-Bench, GenEval, OneIG-Bench, and TIIF Bench testmini. Due to limited space, we only select recent baselines and put detailed results to Tabs. 12–16.

Type	Model	DPG-Bench	GenEval	OneIG-EN	OneIG-ZH	TIIF-Short	TIIF-Long
Proprietary	Seedream 3.0 [25]	88.27	0.84	0.530	0.528	86.02	84.31
	GPT Image 1 [54]	85.15	0.84	0.533	0.474	89.15	88.29
Diffusion	Qwen-Image [67]	88.32	0.87	0.539	0.548	86.14	86.83
	Z-Image [68]	88.14	0.84	0.546	0.535	80.20	83.04
Autoregressive	Janus-Pro [13]	84.19	0.80	0.267	0.240	66.50	65.02
	NextStep-1 [66]	85.28	0.73	0.418	-	-	-
	GLM-Image [89]	84.78	-	0.528	0.511	81.01	81.02
	BitDance	88.28	0.86	0.532	0.512	79.64	78.12

As shown in Tab. 4, BitDance achieves competitive performance with leading commercial and open-source image generation models on GenEval [27], DPG-Bench [36], OneIG-Bench [8], and TIIF-Bench [74], despite training on fewer than 450M samples. After fine-tuning on 100M high-quality editing samples, it also performs strongly on GEdit-Bench [47] and ImgEdit-Bench [84], as shown in Tab. 5. Additional discussion and full results are provided in Appendix C.

4.4 Ablation Study

We conduct ablation studies on ImageNet [18] 256×256 to analyze the components within BitDance. We use BitDance-B as the backbone. All models are trained for 400 epochs and evaluated using CFG.

Continuous VAE vs. Binary Tokenizer. Tab. 6 shows that continuous VAEs [45, 83] perform notably worse than our binary tokenizer for AR generation, suggesting that unconstrained continuous tokens accumulate substantial errors during generation.

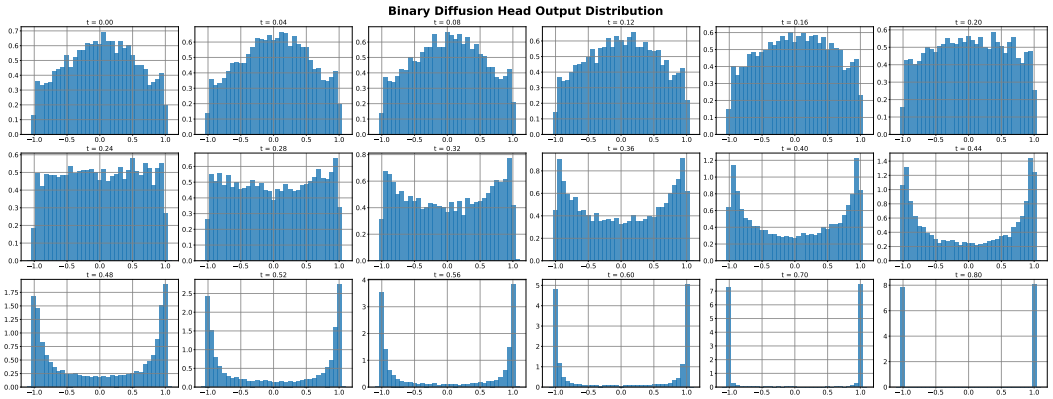


Figure 7: **Output distribution of the binary diffusion head at different timesteps t.** As t increases from 0 to 1 (i.e., as noise decreases), the predictions become sharper and gradually converge to -1 and 1.

Table 5: **Evaluation of image editing** on GEdit-Bench and ImgEdit-Bench. G_SC, G_PQ, and G_O refer to the metrics evaluated by GPT-4.1. Refer to Tab. 19 for full results.

Type	Model	GEdit-Bench-EN (Full set)↑			GEdit-Bench-CN (Full set)↑			ImgEdit-Bench↑
		G_SC	G_PQ	G_O	G_SC	G_PQ	G_O	
Proprietary	Gemini 2.0 [26]	6.73	6.61	6.32	5.43	6.78	5.36	-
	GPT-4o [37]	7.85	7.62	7.53	7.67	7.56	7.30	4.20
Diffusion	OmniGen [77]	5.96	5.89	5.06	-	-	-	2.96
	Step1X-Edit [47]	7.09	6.76	6.70	7.20	6.87	6.86	3.06
	BAGEL [17]	7.36	6.83	6.52	7.34	6.85	6.50	3.42
Autoregressive	NextStep-1 [66]	7.15	7.01	6.58	6.88	7.02	6.40	3.71
	BitDance	7.20	6.78	6.54	7.18	6.82	6.46	3.45

Table 6: **Different tokenizers for AR generation** ($p = 1$).

Tokenizer	FID↓	IS↑
MAR VAE [45]	3.16	289.9
VA-VAE [83]	4.84	273.7
BitDance-Tok	**1.79**	**290.5**

Table 7: **Comparison of different sampling heads** ($p = 1$).

Sampling Head	FID↓	IS↑
Token Cls	OOM	OOM
Bit-wise Cls	8.37	174.5
Binary Diff	**1.79**	**290.5**

Table 8: **Ablation on the design of next-patch diffusion** ($p = 4$).

Model	FID↓	IS↑
Next-Patch Diffusion	**1.98**	**276.7**
Block-wise→Full	2.07	271.8
Patch→Token Raster	2.15	270.0

Sampling Head. We evaluate different sampling heads discussed in Sec. 3.2 in Tab. 7. The token classification head suffers from OOM issues caused by a high volume of parameters, while the bitwise classification head performs poorly due to the restrictive assumption of bit independence.

Next-Patch Diffusion. Two critical designs in next-patch diffusion modeling are investigated: the patch-wise raster scan order for token generation and the use of block causal masks for intra-patch token visibility. As shown in Tab. 8, both designs effectively improve generation performance.

Diffusion Sampling Steps. Fig. 8 shows the impact of sampling steps in the binary diffusion head. Good performance is achieved with as few as 10 steps, indicating that the discrete nature of binary tokens makes sampling easier than for continuous tokens and thus supports fast, robust generation.

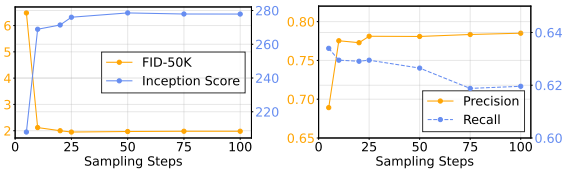


Figure 8: **Sampling steps in binary diffusion head** ($p = 4$). High-quality generation can be achieved with a small number of sampling steps. (e.g., 10-20 steps).

Prediction of Binary Diffusion Head. Our binary diffusion head adopts the x-prediction [44] formulation in Sec. 3.2, directly predicting the clean binary latents. As shown in Fig. 7, when t is small and noise is high, many predictions cluster around 0. As t increases, the predictions become sharper and converge to -1 and 1, indicating that the head implicitly learns a binary discrete distribution without hand-crafted architectural or objective constraints.

5 Conclusion

We present BitDance, a simple and scalable autoregressive model for high-quality image generation via efficient prediction of binary visual tokens. BitDance is the first to scale a visual tokenizer vocabulary to 2^{256}, achieving reconstruction fidelity comparable to continuous VAEs. With the proposed binary diffusion head and next-patch diffusion modeling, BitDance effectively captures the joint distribution of multiple binary tokens, enabling efficient and accurate parallel prediction. Extensive experiments across diverse generative tasks demonstrate its effectiveness and efficiency.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (No. 62306261), HK RGC-Early Career Scheme (No. 24211525) and ITSP Platform Project (No. ITS/600/24FP). This study was supported in part by the Centre for Perceptual and Interactive Intelligence, a CUHK-led InnoCentre under the InnoHK initiative of the Innovation and Technology Commission of the Hong Kong Special Administrative Region Government. This work is also partially supported by Hong Kong RGC Strategic Topics Grant (No. STG1/E-403/24-N and No. STG1/M-404/26-N), and CUHK-CUHK(SZ)-GDST Joint Collaboration Fund (No. YSP26-4760949).

This work is also partially supported by the National Natural Science Foundation of China (Grant Nos. 62576342, 62550062, 62425606, 32341009) and the Beijing Natural Science Foundation (Grant Nos. L252145, 4252054).

References

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

[2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025. 7

[3] Yuang Ai, Qihang Fan, Xuefeng Hu, Zhenheng Yang, Ran He, and Huaibo Huang. Dico: Revitalizing convnets for scalable and efficient diffusion modeling. In *NeurIPS*, 2025. 8, 15

[4] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn.openai.com/papers/dall-e-3.pdf*, 2023. 19

[5] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 20

[6] Qi Cai, Yehao Li, Yingwei Pan, Ting Yao, and Tao Mei. Hidream-i1: An open-source high-efficient image generative foundation model. In *ACM MM*, pages 13636–13639, 2025. 17, 18, 19

[7] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *CVPR*, 2022. 4, 8, 15

[8] Jingjing Chang, Yixiao Fang, Peng Xing, Shuhan Wu, Wei Cheng, Rui Wang, Xianfang Zeng, Gang Yu, and Hai-Bao Chen. Oneig-bench: Omni-dimensional nuanced evaluation for image generation. In *NeurIPS*, 2025. 3, 8, 18, 19

[9] Junyu Chen, Han Cai, Junsong Chen, Enze Xie, Shang Yang, Haotian Tang, Muyang Li, Yao Lu, and Song Han. Deep compression autoencoder for efficient high-resolution diffusion models. *arXiv preprint arXiv:2410.10733*, 2024. 7

[10] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *ECCV*, 2024. 19

[11] Junsong Chen, YU Jincheng, GE Chongjian, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024. 17, 18

[12] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 14

[13] 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*, 2025. 8, 17, 18, 19

[14] 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*, 2025. 19

[15] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with ad-

vanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 1

[16] Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, et al. Emu3. 5: Native multimodal models are world learners. *arXiv preprint arXiv:2510.26583*, 2025. 3

[17] 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. 9, 17, 18, 19, 20

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 8, 14

[19] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 14

[20] 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 *ICML*, 2024. 17, 18, 19

[21] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 2, 14

[22] Lijie Fan, Tianhong Li, Siyang Qin, Yuanzhen Li, Chen Sun, Michael Rubinstein, Deqing Sun, Kaiming He, and Yonglong Tian. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens. In *ICLR*, 2025. 2

[23] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. In *NeurIPS*, 2023. 14

[24] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Mdtv2: Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023. 15

[25] 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. 8, 16, 17, 18, 19

[26] Google Gemini2. Experiment with gemini 2.0 flash native image generation, 2025. 9, 20

[27] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2023. 3, 8, 17, 18

[28] Google. Imagen 4 model card. https://storage.googleapis.com/deepmind-media/Model-Cards/Imagen-4-Model-Card.pdf, 2025. 18

[29] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 1

[30] 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. In *NeurIPS*, 2025. 17, 18

[31] 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 *CVPR*, 2025. 3, 4, 5, 7, 17, 18, 19

[32] Yefei He, Yuanyu He, Shaoxuan He, Feng Chen, Hong Zhou, Kaipeng Zhang, and Bohan Zhuang. Neighboring autoregressive modeling for efficient visual generation. *arXiv preprint arXiv:2503.10696*, 2025. 8, 15

[33] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 14

[34] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3, 14

[35] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 17

[36] 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. 3, 8, 17

[37] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 9, 20

[38] Guolin Ke and Hui Xue. Hyperspherical latents improve continuous-token autoregressive generation. In *ICLR*, 2026. 2, 3, 5, 8, 14, 15

[39] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 14

[40] Kuaishou Kolors Team. Kolors 2.0. https://app.klingai.com/cn/, 2025. 19

[41] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019. 14

[42] Black Forest Labs. Flux, 2024. 14, 17, 18, 19

[43] Haopeng Li, Jinyue Yang, Guoqi Li, and Huan Wang. Autoregressive image generation with randomized parallel decoding. *arXiv preprint arXiv:2503.10568*, 2025. 3, 4, 6

[44] Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025. 5, 9

[45] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. In *NeurIPS*, 2024. 2, 3, 4, 7, 8, 9, 14, 15

[46] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, et al. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024. 19

[47] Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025. 8, 9, 20

[48] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 5

[49] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 14, 17

[50] 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. 7, 8, 14, 15

[51] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *ECCV*, 2024. 15

[52] Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Liang Zhao, et al. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation. *arXiv preprint arXiv:2411.07975*, 2024. 18

[53] Midjourney. Midjourney v7. https://www.midjourney.com/home, 2025. 19

[54] OpenAI. Gpt-image-1. https://openai.com/zh-Hans-CN/index/introducing-4o-image-generation/, 2025. 8, 17, 18, 19

[55] 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 *CVPR*, 2025. 3, 4, 6, 7, 8, 15

[56] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 7, 8, 14, 15

[57] Qi Qin, Le Zhuo, Yi Xin, Ruoyi Du, Zhen Li, Bin Fu, Yiting Lu, Jiakang Yuan, Xinyue Li, Dongyang Liu, et al. Lumina-image 2.0: A unified and efficient image generative framework. *arXiv preprint arXiv:2503.21758*, 2025. 17, 18

[58] Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Beyond next-token: Next-x prediction for autoregressive visual generation. In *ICCV*, 2025. 2, 3, 8, 15

[59] Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. FlowAR: Scale-wise autoregressive image generation meets flow matching. In *ICML*, 2025. 3

[60] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 7, 14

[61] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016. 14

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022. 16

[63] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024. 16

[64] 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. 2, 3, 7, 8, 14, 15

[65] Yutao Sun and et al. Multimodal latent language modeling with next-token diffusion. *arXiv preprint arXiv:2412.08635*, 2024. 3

[66] NextStep Team, Chunrui Han, Guopeng Li, Jingwei Wu, Quan Sun, Yan Cai, Yuang Peng, Zheng Ge, Deyu Zhou, Haomiao Tang, et al. Nextstep-1: Toward autoregressive image generation with continuous tokens at scale. In *ICLR*, 2026. 2, 3, 5, 8, 9, 16, 17, 18, 19, 20

[67] Qwen-Image Team. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025. 8, 17, 18, 19

[68] Z-Image Team. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025. 8, 16, 17, 18, 19

[69] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *NeurIPS*, 2024. 2, 3, 4, 8, 14, 15

[70] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. In *NeurIPS*, 2017. 2, 14

[71] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 2, 3, 16, 17, 18

[72] Yuqing Wang, Zhijie Lin, Yao Teng, Yuanzhi Zhu, Shuhuai Ren, Jiashi Feng, and Xihui Liu. Bridging continuous and discrete tokens for autoregressive visual generation. *arXiv preprint arXiv:2503.16430*, 2025. 3

[73] Yuqing Wang, Shuhuai Ren, Zhijie Lin, Yujin Han, Haoyuan Guo, Zhenheng Yang, Difan Zou, Jiashi Feng, and Xihui Liu. Parallelized autoregressive visual generation. In *CVPR*, 2025. 3, 4, 6, 8, 15
[74] Xinyu Wei, Jinrui Zhang, Zeqing Wang, Hongyang Wei, Zhen Guo, and Lei Zhang. Tiif-bench: How does your t2i model follow your instructions? *arXiv preprint arXiv:2506.02161*, 2025. 8, 17, 19
[75] Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025. 18
[76] Xianfeng Wu, Yajing Bai, Haoze Zheng, Harold Haodong Chen, Yexin Liu, Zihao Wang, Xuran Ma, Wen-Jie Shu, Xianzu Wu, Harry Yang, et al. Lightgen: Efficient image generation through knowledge distillation and direct preference optimization. *arXiv preprint arXiv:2503.08619*, 2025. 19
[77] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. *arXiv preprint arXiv:2409.11340*, 2024. 9, 20
[78] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *ICLR*, 2025. 7
[79] Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng Yu, Ligeng Zhu, Chengyue Wu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, et al. Sana 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. *arXiv preprint arXiv:2501.18427*, 2025. 14, 18, 19
[80] 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 *ICLR*, 2025. 7
[81] Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025. 18
[82] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 7, 16
[83] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *CVPR*, 2025. 8, 9
[84] Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025. 8, 20
[85] Lijun Yu, Jose Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion-tokenizer is key to visual generation. In *ICLR*, 2024. 3, 4, 8, 14, 15
[86] Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. *arXiv preprint arXiv:2411.15738*, 2024. 20
[87] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In *ICCV*, 2025. 3, 8, 15
[88] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *ICLR*, 2025. 8, 15
[89] Z.ai. Glm-image: Auto-regressive for dense-knowledge and high-fidelity image generation. https://z.ai/blog/glm-image, 2026. 2, 8, 17, 18, 19
[90] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *NeurIPS*, 2023. 20
[91] Yue Zhao, Yuanjun Xiong, and Philipp Krähenbühl. Image and video tokenization with binary spherical quantization. In *ICLR*, 2025. 3, 14
[92] Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025. 8, 15
[93] Wendi Zheng, Jiayan Teng, Zhuoyi Yang, Weihan Wang, Jidong Chen, Xiaotao Gu, Yuxiao Dong, Ming Ding, and Jie Tang. Cogview3: Finer and faster text-to-image generation via relay diffusion. *arXiv preprint arXiv:2403.05121*, 2024. 19
[94] Shaobin Zhuang, Yiwei Guo, Canmiao Fu, Zhipeng Huang, Zeyue Tian, Fangyikang Wang, Ying Zhang, Chen Li, and Yali Wang. Wetok: Powerful discrete tokenization for high-fidelity visual reconstruction. In *ICLR*, 2026. 4, 7, 14
[95] Le Zhuo, Ruoyi Du, Han Xiao, Yangguang Li, Dongyang Liu, Rongjie Huang, Wenze Liu, Xiangyang Zhu, Fu-Yun Wang, Zhanyu Ma, et al. Lumina-next: Making lumina-t2x stronger and faster with next-dit. In *NeurIPS*, 2024. 19

Appendix

We provide the following supplementary information in the Appendix:

- A. additional details of our binary visual tokenizer.
- B. additional details of class-conditional image generation on ImageNet.
- C. additional details of text-to-image generation and image editing.
- D. additional qualitative results of BitDance models.
- E. discussion of limitations and broader impacts of this work.

A Additional Details of Visual Tokenizer

Related Work. To mitigate the prohibitive costs of pixel-space training [12, 34], Variational Autoencoders (VAEs) [39] are widely used to project visual content into continuous latent spaces. Due to their high-fidelity reconstruction and smooth gradient flow, VAEs have become the standard for leading diffusion models [60, 56, 42, 79]. Conversely, discrete tokenizers using Vector Quantization (VQ) [70, 21] often struggle with quantization errors and instability of codebook utilization. Recent works have increasingly investigated binary quantization as a solution. MAGVIT-v2 [85] introduces Lookup-Free Quantization (LFQ) to scale vocabularies to 2^{18}, but its entropy loss incurs linear memory costs that hinder further scaling. Subsequent works like BSQ [91] and WeTok [94] attempt to resolve this via independence assumptions or grouping strategies, respectively. Building on these advances, we aim to further explore scaling up the vocabulary size to achieve higher token entropy.

Training Details of BitDance-Tok. We utilize DataComp-1B [23] as our main training set, supplemented with high-quality face and text datasets to improve domain-specific reconstruction. Training is conducted at a 256×256 resolution for 400K steps using a batch size of 1024. We train the model using the Adam optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.9$, while setting the weight decay to 0. We employ an exponential moving average (EMA) of model weights during training, with a decay rate of 0.999. To support large-vocabulary training, we compute the entropy loss by grouping channels [94], where each group consists of 8 channels. Following previous works [50, 94], the tokenizer adopts a pure CNN architecture, enabling generalization on various input resolutions.

B Additional Details of Class-conditional Image Generation

Datasets and Metrics. We follow prior studies [45, 69, 64] and evaluate our models on the class-conditional ImageNet-1K [18] generation benchmark at a resolution of 256×256. The main evaluation metric is the Fréchet Inception Distance (FID) [33]. Additionally, we report Inception Score (IS) [61] as well as Precision and Recall [41] as complementary measures of generative quality. All evaluation metrics are computed using OpenAI's TensorFlow-based evaluation toolkit [19].

Implementation Details. Following MAR [45], we explore the scaling capabilities of three BitDance models of varying sizes. BitDance-B, -L, and -H feature 24, 32, and 40 Transformer blocks with widths of 768, 1024, and 1280, respectively. The binary diffusion head maintains a consistent width, while the number of blocks is set at 6, 8, and 12. Furthermore, our model nomenclature incorporates the patch size p used in next-patch diffusion modeling. For instance, BitDance-B-4x denotes a patch size of $p = 2$, where each AR step generates $p^2 = 4$ tokens. Following previous works [45, 38, 69], we conduct our experiments using a 16× downsampling binary tokenizer trained on ImageNet. The AR model is trained for 800 epochs using the AdamW [49] optimizer with $\beta = (0.9, 0.95)$ and a weight decay of 0.05. We employ a batch size of 1024 and a cosine learning rate schedule with 20K warmup steps. Additionally, an Exponential Moving Average (EMA) decay rate of 0.9999 is applied.

Full Comparison Results. Due to space constraints in the main text, we provide the full comparison results for class-conditional image generation in Tab. 9 and Tab. 10.

Table 9: **Evaluation of class-conditional image generation on ImageNet 256×256.** We mark models using additional self-supervised models (e.g., DINO) in gray.

Model	Type	Order	Tokenizer	#Params	FID↓	IS↑	Pre.↑	Rec.↑
Continuous Tokens								
DiT-XL/2 [56]	Diff.	-	VAE	675M	2.27	278.2	0.83	0.57
SiT-XL/2 [51]	Diff.	-	VAE	675M	2.06	277.5	0.83	0.59
DiCo-XL [3]	Diff.	-	VAE	701M	2.05	282.2	0.83	0.59
MDTv2 [24]	Mask.+Diff.	-	VAE	675M	1.58	314.7	0.79	0.65
REPA [88]	Diff.	-	VAE	675M	1.42	305.7	0.80	0.65
RAE [92]	Diff.	-	RAE	675M	1.13	262.6	0.78	0.67
MAR-B [45]	Mask.	random	VAE	208M	2.31	281.7	0.82	0.57
MAR-L [45]	Mask.	random	VAE	479M	1.78	296.0	0.81	0.60
MAR-H [45]	Mask.	random	VAE	943M	1.55	303.7	0.81	0.62
xAR-B [58]	AR+Diff.	raster	VAE	172M	1.72	280.4	0.82	0.59
xAR-L [58]	AR+Diff.	raster	VAE	608M	1.28	292.5	0.82	0.62
xAR-H [58]	AR+Diff.	raster	VAE	1.1B	1.24	301.6	0.83	0.64
SphereAR-B [38]	AR	raster	VAE	208M	1.92	277.8	0.81	0.61
SphereAR-L [38]	AR	raster	VAE	479M	1.54	295.9	0.80	0.63
SphereAR-H [38]	AR	raster	VAE	943M	1.34	300.0	0.80	0.64
Discrete Tokens								
LlamaGen-L [64]	AR	raster	VQ	343M	3.07	256.1	0.83	0.52
LlamaGen-XL [64]	AR	raster	VQ	775M	2.62	244.1	0.80	0.57
LlamaGen-XXL [64]	AR	raster	VQ	1.4B	2.34	253.9	0.80	0.59
RandAR-L [55]	AR	random	VQ	343M	2.55	288.8	0.81	0.58
RandAR-XL [55]	AR	random	VQ	775M	2.22	314.2	0.80	0.60
RandAR-XXL [55]	AR	random	VQ	1.4B	2.15	322.0	0.79	0.62
RAR-L [87]	AR	hybrid	VQ	461M	1.70	299.5	0.81	0.60
RAR-XL [87]	AR	hybrid	VQ	955M	1.50	306.9	0.80	0.62
RAR-XXL [87]	AR	hybrid	VQ	1.5B	1.48	**326.0**	0.80	0.63
OpenMAGVIT2-XL [50]	AR	raster	LFQ	804M	2.51	271.7	**0.84**	0.54
MAGVIT-v2 [85]	Mask.	random	LFQ	307M	1.78	319.4	-	-
VAR-d20 [69]	VAR	-	VQ	600M	2.57	302.6	0.83	0.56
VAR-d30 [69]	VAR	-	VQ	2B	1.92	323.1	0.82	0.59
BitDance-B-1x	AR	raster	LFQ	242M	1.68	297.1	0.80	0.62
BitDance-L-1x	AR	raster	LFQ	527M	1.31	300.2	0.80	0.64
BitDance-H-1x	AR	raster	LFQ	1.0B	**1.24**	304.4	0.81	**0.64**

Table 10: **Overall comparison with parallel image generation methods on ImageNet 256×256.** Throughput is measured on one A100 with a batch size of 64 at bfloat16 precision.

Model	Type	Order	#Params	Steps	Throughput↑	FID↓	IS↑	Pre.↑	Rec.↑
DiT-XL/2 [56]	Diff.	-	675M	250	1.06 img/s	2.27	278.2	0.83	0.57
DiCo-XL [3]	Diff.	-	701M	250	2.62 img/s	2.05	282.2	0.83	0.59
MaskGIT [7]	Mask.	random	227M	8	50.73 img/s	6.18	182.1	0.80	0.51
MAR-B [45]	Mask.	random	208M	256	1.83 img/s	2.31	281.7	0.82	0.57
MAR-L [45]	Mask.	random	479M	256	1.39 img/s	1.78	296.0	0.81	0.60
VAR-d20 [69]	VAR	-	600M	10	71.31 img/s	2.57	302.6	**0.83**	0.56
VAR-d24 [69]	VAR	-	1.0B	10	47.22 img/s	2.09	312.9	0.82	0.59
PAR-L [73]	AR	hybrid	343M	147	15.01 img/s	3.76	218.9	0.81	0.60
PAR-XL [73]	AR	hybrid	775M	147	8.09 img/s	2.61	259.2	0.80	0.62
PAR-XXL [73]	AR	hybrid	1.4B	147	5.17 img/s	2.35	263.2	0.80	0.62
NAR-L [32]	AR	hybrid	372M	31	40.03 img/s	3.06	263.9	0.81	0.53
NAR-XL [32]	AR	hybrid	816M	31	23.12 img/s	2.70	277.5	0.81	0.58
NAR-XXL [32]	AR	hybrid	1.5B	31	15.37 img/s	2.58	293.5	0.82	0.57
RandAR-L [55]	AR	random	343M	88	25.12 img/s	2.55	288.8	0.81	0.58
RandAR-XL [55]	AR	random	775M	88	16.01 img/s	2.25	317.8	0.80	0.60
RandAR-XXL [55]	AR	random	1.4B	88	10.39 img/s	2.15	**322.0**	0.79	0.62
BitDance-B-4x	AR	raster	260M	64	24.18 img/s	**1.69**	291.2	0.79	**0.63**
BitDance-B-16x	AR	raster	260M	16	**90.26** img/s	1.91	283.8	0.78	0.62

Table 11: **Training recipe of BitDance for text-to-image generation.** In the adaptation phase, which requires only a few training steps, we transition a model with 16-token parallel prediction ($p = 4$) into a model with 64-token parallel prediction ($p = 8$) to achieve faster inference speeds.

Stage	**PT**	**CT**	**SFT**	**Adapt (optional)**
Learning rate	1.0×10^{-4}	1.0×10^{-4}	2.0×10^{-5}	2.0×10^{-5}
LR scheduler	Constant	Constant	Constant	Constant
Weight decay	0.0	0.0	0.0	0.0
Gradient norm clip	1.0	1.0	1.0	1.0
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 1.0 \times 10^{-15}$)			
Loss weight (text : vision)	0.01 : 1	0.01 : 1	0.01 : 1	0.01 : 1
Warm-up steps	2000	1000	500	500
Training steps	100K	40K	40K	20K
Global batch size	2560	2480	2320	1536
# Training samples	256M	99.2M	92.8M	30.7M
Image resolution (256px : 512px : 1024px)	2 : 7 : 1	0 : 1 : 1	0 : 1 : 1	0 : 0 : 1
Text token drop prob	0.1	0.1	0.1	0.1

C Additional Details of Text-to-Image Generation and Editing

C.1 Model Design

To scale up BitDance on text-to-image generation and image editing tasks, we choose the pretrained Qwen3-14B [82] as the base AR model. Following previous works [66, 71], the AR model serves as both the text encoder and image generator. Our primary goal is to enhance it with image generation capability using high-quality image-text pairs. To balance reconstruction quality and generative convergence speed, we choose the tokenizer with 16× downsampling ratio for our experiments. To achieve more efficient high-resolution generation, we set the patch size p to 4 and enable parallel prediction of $p^2 = 16$ tokens per step. For positional encoding, in addition to the 1D RoPE [63] used in LLM, we incorporate 2D sinusoidal positional embeddings to enhance the model's spatial awareness of images.

Notably, to support **varying image resolution and aspect ratios**, we add two resolution tokens `[res_i]` and `[res_j]` at the beginning of vision tokens:

```
[bos], {text tokens}, [boi], [res_i], [res_j], {vision tokens}, [eoi], [eos],
```

where i and j indicate the number of vision tokens on the height and width dimensions. In this way, we can decide how many tokens (i.e., $i \times j$) to be decoded during inference.

C.2 Training Recipe

For text-to-image generation, we adopt a three-stage training pipeline, including **pre-training (PT)**, **continued training (CT)** and **supervised fine-tuning (SFT)**. Our PT and CT data is collected from open-sourced datasets such as LAION [62]. The SFT data is collected from both open-source datasets and a small number of images generated by other text-to-image models such as Seedream [25] and Z-Image-Turbo [68], which help our model to quickly adapt to the distribution of high-quality images and achieve strong visual fidelity with substantially fewer training samples. Besides, we also preserve a small amount of image-to-text data to maintain the text understanding ability of LLM.

To further accelerate model inference, we introduce an additional **adaptation stage** that transitions the SFT model from 16-token parallel prediction to 64-token parallel prediction. Specifically, we inherit the weights from the SFT model and fine-tune it using a small amount of high-quality data. We find that the model can adapt to predicting more tokens in parallel with only a few training steps.

The training parameters are listed in Tab. 11. Different from prior works, we adopt a **mixed-resolution training strategy** during pre-training, which we find crucial for training stability. Specifically, we use 512px as the primary training resolution, while jointly incorporating 256px images to improve training throughput and 1024px images to enhance stability and robustness at higher resolutions. This mixed-resolution setup enables stable optimization while maintaining both efficiency and high-

Table 12: **Evaluation of text-to-image generation on DPG-Bench [36].** We mark the best and second-best performance among autoregressive models in **bold** and underline, respectively.

Model	Global	Entity	Attribute	Relation	Other	Overall↑
Proprietary Models						
GPT Image 1 [54]	88.89	88.94	89.84	92.63	90.96	85.15
Seedream 3.0 [25]	94.31	92.65	91.36	92.78	88.24	88.27
Diffusion Models						
PixArt-α [11]	86.89	82.89	88.94	86.59	87.68	80.54
FLUX.1-Dev [42]	74.35	90.00	88.96	90.87	88.33	83.84
SD3 Medium [20]	87.90	91.01	88.83	80.70	88.68	84.08
Z-Image-Turbo [68]	91.29	89.59	90.14	92.16	88.68	84.86
BAGEL [17]	-	-	-	-	-	85.07
HiDream-I1-Full [6]	76.44	90.22	89.48	93.74	91.83	85.89
Lumina-Image-2.0 [57]	-	91.97	90.20	94.85	-	87.20
Z-Image [68]	93.39	91.22	93.16	92.22	91.52	88.14
Qwen-Image [67]	91.32	91.56	92.02	94.31	92.73	88.32
Autoregressive Models						
Emu3-Gen [71]	85.21	86.68	86.84	90.22	83.15	80.60
Infinity [31]	**93.11**	-	-	90.76	-	83.46
Janus-Pro [13]	86.90	88.90	89.40	89.32	89.48	84.19
Tar [30]	83.98	88.62	88.05	**93.98**	84.86	84.19
NextStep-1 [66]	-	-	-	-	-	85.28
GLM-Image [89]	87.74	90.25	89.08	92.15	90.17	84.78
BitDance	89.53	**93.76**	**92.47**	91.81	**90.26**	**88.28**

resolution generalization. During the continued training stage, we increase the proportion of 1024px images to enhance the model's performance in high-resolution image generation.

For all training stages, we use the AdamW [49] optimizer with $\beta = (0.9, 0.95)$ and a constant learning rate. The visual tokenizer is frozen, while all other model parameters remain trainable. The **training loss** consists of two parts: a **vision loss** from the binary diffusion head and a **text loss** (cross-entropy loss) from the AR model. We set the relative weighting of vision and text loss as $1 : 0.01$, which allows the AR model to maintain fundamental text encoding capabilities while keeping the primary focus on image generation. To enable classifier-free guidance [35] and improve model robustness, we randomly drop text tokens with a predefined token dropout probability of 0.1 during training. This encourages the model to rely on both text and image cues and prevents overfitting to the textual input.

After training the text-to-image model, we further fine-tune it for **image editing** using 100M high-quality editing samples. The training hyperparameters are kept identical to those used in the SFT stage of text-to-image model training.

C.3 Main Results

Text-to-Image Generation. We conduct a comprehensive evaluation of BitDance on text-to-image generation, covering capabilities such as prompt following, text rendering, and reasoning. We compare it against state-of-the-art proprietary models (e.g., Seeddream [25], GPT Image 1 [54]), diffusion models (e.g., Qwen-Image [67], Z-Image [68]), and autoregressive models (e.g., GLM-Image [89], NextStep-1 [66]). The results demonstrate that BitDance achieves state-of-the-art performance among autoregressive models and is comparable to leading proprietary and diffusion models.

As shown in Tab. 12, Tab. 13 and Tab. 16, BitDance demonstrates strong prompt-following capabilities. On GenEval [27] and DPG-Bench [36], BitDance outperforms the majority of existing open-source models across most evaluation dimensions. BitDance achieves an overall GenEval score of 0.86 and DPG-Bench score of 88.28, ranking among the top-performing methods. On the TIIF benchmark (testmini) [74], BitDance achieves the second-best overall performance among autoregressive models, demonstrating its strong capability in executing complex and diverse user instructions. Importantly, these competitive results are achieved with **significantly fewer training data.** BitDance is trained on fewer than 450M image–text pairs, which is orders of magnitude smaller than the billion- to multi-

Table 13: **Evaluation of text-to-image generation on GenEval [27].** We mark the best and second-best performance among autoregressive models in **bold** and underline, respectively.

Model	Single Obj.	Two Obj.	Count	Colors	Pos.	Color Attri.	Overall↑
Proprietary Models							
GPT Image 1 [54]	0.99	0.92	0.85	0.92	0.75	0.61	0.84
Seedream 3.0 [25]	0.99	0.96	0.91	0.93	0.47	0.80	0.84
Diffusion Models							
PixArt-α [11]	0.98	0.50	0.44	0.80	0.08	0.07	0.48
SD3 Medium [20]	0.98	0.74	0.63	0.67	0.34	0.36	0.62
JanusFlow [52]	0.97	0.59	0.45	0.83	0.53	0.42	0.63
FLUX.1-Dev [42]	0.98	0.81	0.74	0.79	0.22	0.45	0.66
SD3.5-Large [20]	0.98	0.89	0.73	0.83	0.34	0.47	0.71
Lumina-Image-2.0 [57]	-	0.87	0.67	-	-	0.62	0.73
Show-o2 [81]	1.00	0.87	0.58	0.92	0.52	0.62	0.76
Z-Image-Turbo [68]	1.00	0.95	0.77	0.89	0.65	0.68	0.82
HiDream-I1-Full [6]	1.00	0.98	0.79	0.91	0.60	0.72	0.83
Z-Image [68]	1.00	0.94	0.78	0.93	0.62	0.77	0.84
Qwen-Image [67]	0.99	0.92	0.89	0.88	0.76	0.77	0.87
BAGEL [17]	0.98	0.95	0.84	0.95	0.78	0.77	0.88
Autoregressive Models							
Emu3-Gen [71]	0.98	0.71	0.34	0.81	0.17	0.21	0.54
Infinity [31]	-	0.85	-	-	0.49	0.57	0.73
Janus-Pro [13]	0.99	0.89	0.59	0.90	0.79	0.66	0.80
Tar [30]	0.98	0.92	**0.83**	0.85	**0.80**	0.65	0.84
NextStep-1 [66]	-	-	-	-	-	-	0.73
BitDance	**1.00**	**0.96**	0.71	**0.95**	0.72	**0.83**	**0.86**

Table 14: **Evaluation of text-to-image generation on OneIG-EN [8].** We mark the best and second-best performance among autoregressive models in **bold** and underline, respectively.

Model	Alignment	Text	Reasoning	Style	Diversity	Overall↑
Proprietary Models						
Imagen 4 [28]	0.857	0.805	0.338	0.377	0.199	0.515
Seedream 3.0 [25]	0.818	0.865	0.275	0.413	0.277	0.530
GPT Image 1 [54]	0.851	0.857	0.345	0.462	0.151	0.533
Diffusion Models						
Show-o2 [81]	0.817	0.002	0.226	0.317	0.177	0.308
SANA-1.5 [79]	0.765	0.069	0.217	0.401	0.216	0.334
BAGEL [17]	0.769	0.244	0.173	0.367	0.251	0.361
FLUX.1-Dev [42]	0.786	0.523	0.253	0.368	0.238	0.434
OmniGen2 [75]	0.804	0.680	0.271	0.377	0.242	0.475
HiDream-I1-Full [6]	0.829	0.707	0.317	0.347	0.186	0.477
Z-Image-Turbo [68]	0.840	0.994	0.298	0.368	0.139	0.528
Qwen-Image [67]	0.882	0.891	0.306	0.418	0.197	0.539
Z-Image [68]	0.881	0.987	0.280	0.387	0.194	0.546
Autoregressive Models						
Janus-Pro [13]	0.553	0.001	0.139	0.276	**0.365**	0.267
NextStep-1 [66]	0.826	0.507	0.224	0.332	0.199	0.418
GLM-Image [89]	0.805	**0.969**	**0.298**	0.353	0.213	0.528
BitDance	**0.853**	0.937	0.297	**0.395**	0.177	**0.532**

billion-scale datasets typically used by current leading commercial models. Despite this substantial data disadvantage, BitDance is able to match or closely approach the performance of such proprietary systems on multiple benchmarks.

Results on OneIG Bench [8] further validate the data efficiency and generalization ability of BitDance. As shown in Tab. 14 and Tab. 15, on both OneIG-EN and OneIG-ZH, BitDance achieves competitive average scores, narrowing the gap with state-of-the-art commercial models, particularly in text fidelity and alignment. Taken together, these results highlight that BitDance effectively bridges the

Table 15: **Evaluation of text-to-image generation on OneIG-ZH [8].** We mark the best and second-best performance among autoregressive models in **bold** and underline, respectively.

Model	Alignment	Text	Reasoning	Style	Diversity	Overall↑
Proprietary Models						
Kolors 2.0 [40]	0.738	0.502	0.226	0.331	0.333	0.426
GPT Image 1 [54]	0.812	0.650	0.300	0.449	0.159	0.474
Seedream 3.0 [25]	0.793	0.928	0.281	0.397	0.243	0.528
Diffusion Models						
HiDream-I1-Full [6]	0.620	0.205	0.256	0.304	0.300	0.337
CogView4 [93]	0.700	0.193	0.236	0.348	0.214	0.338
BAGEL [17]	0.672	0.365	0.186	0.357	0.268	0.370
Z-Image-Turbo [68]	0.782	0.982	0.276	0.361	0.134	0.507
Qwen-Image [67]	0.825	0.963	0.267	0.405	0.279	0.548
Z-Image [68]	0.793	0.988	0.266	0.386	0.243	0.535
Autoregressive Models						
Janus-Pro [13]	0.324	0.148	0.104	0.264	**0.358**	0.240
GLM-Image [89]	0.738	**0.976**	**0.284**	0.335	0.221	0.511
BitDance	**0.786**	0.961	0.276	**0.376**	0.159	**0.512**

Table 16: **Evaluation of text-to-image generation on TIIF Bench testmini [74].** We mark the best and second-best performance among autoregressive models in **bold** and underline, respectively.

Model	Overall↑		Basic Following								Advanced Following												Designer	
			Avg		Attribute		Relation		Reasoning		Avg		Attr.+Rela.		Attr.+Reas.		Rela.+Reas.		Style		Text		Real World	
	short	long	short	long	short	long	short	long	short	long	short	long	short	long	short	long	short	long	short	long	short	long	short	long
Proprietary Models																								
Midjourney V7 [53]	68.74	65.69	77.41	76.00	77.58	81.83	82.07	76.82	72.57	69.32	64.66	60.53	67.20	62.70	81.22	71.59	60.72	64.59	83.33	80.00	24.83	20.83	68.83	63.61
DALL-E 3 [4]	74.96	70.81	78.72	78.50	79.50	79.83	80.82	78.82	75.82	76.82	73.39	67.27	73.45	67.20	72.01	71.34	63.59	60.72	89.66	86.67	66.83	54.83	72.93	60.99
Seedream 3.0 [25]	86.02	84.31	87.07	84.93	90.50	90.00	89.85	85.94	80.86	78.86	79.16	80.60	79.76	81.82	77.23	78.85	75.64	78.64	100.0	93.33	97.17	87.78	83.21	83.58
GPT Image 1 [54]	89.15	88.29	90.75	89.66	91.33	87.08	84.57	84.57	96.32	97.32	88.55	88.35	87.07	89.44	87.22	83.96	85.59	83.21	90.00	93.33	89.83	86.83	89.73	93.46
Diffusion Models																								
Lumina-Next [95]	50.93	52.46	64.58	66.08	56.83	59.33	67.57	71.82	69.32	67.07	44.75	45.63	51.44	43.20	51.09	59.72	44.72	54.46	70.00	66.67	0.00	0.83	47.56	49.05
Hunyuan-DiT [46]	51.38	53.28	69.33	69.00	65.83	69.83	78.07	73.82	64.07	63.32	42.62	45.45	50.20	41.57	59.22	61.84	47.84	51.09	56.67	73.33	0.00	0.83	40.10	44.20
PixArt-Σ [10]	62.00	58.12	70.66	75.25	69.33	78.83	75.07	77.32	67.57	69.57	57.65	49.50	65.20	56.57	66.96	61.72	66.59	54.59	83.33	70.00	1.83	1.83	62.11	52.41
SANA 1.5 [79]	67.15	65.73	79.66	77.08	79.83	77.83	85.57	83.57	73.57	69.82	61.50	60.67	65.32	56.57	69.96	73.09	62.96	65.84	80.00	80.00	17.83	15.83	71.07	68.83
SD 3 [20]	67.46	66.09	78.32	77.75	83.33	79.83	82.07	78.82	71.07	74.07	61.46	59.56	61.07	64.07	68.84	70.34	50.96	57.84	66.67	76.67	59.83	20.83	63.23	67.34
FLUX.1-dev [42]	71.09	71.78	83.12	78.65	87.05	83.17	87.25	80.39	75.01	72.39	65.79	68.54	67.07	73.69	73.84	73.34	69.09	71.59	66.67	66.67	43.83	52.83	70.72	71.47
Z-Image-Turbo [68]	77.73	80.05	81.85	81.59	86.50	87.00	82.88	79.99	76.17	77.77	68.32	74.69	72.04	75.24	60.22	73.33	68.90	71.92	83.33	93.33	83.71	84.62	85.82	77.24
Z-Image [68]	80.20	83.04	78.36	82.79	79.50	86.50	80.45	79.94	75.13	81.94	72.89	77.02	72.91	77.56	66.99	73.82	73.89	75.62	90.00	93.33	94.84	93.21	88.06	85.45
Qwen-Image [67]	86.14	86.83	90.18	87.22	90.50	91.50	88.22	90.78	79.81	79.38	79.30	80.88	79.21	78.94	78.85	81.69	75.57	78.59	100.0	100.0	92.76	89.14	90.30	91.42
Autoregressive Models																								
LightGen [76]	53.22	43.41	66.58	47.91	55.83	47.33	74.82	45.82	69.07	50.57	46.74	41.53	62.44	40.82	61.71	50.47	50.34	45.34	53.33	53.33	0.00	6.83	50.92	50.55
Infinity [31]	62.07	62.32	73.08	75.41	74.33	76.83	72.82	77.57	72.07	71.82	56.64	54.98	60.44	55.57	**74.22**	64.71	60.22	59.71	80.00	73.33	10.83	23.83	54.28	56.89
Janus-Pro [14]	66.50	65.02	**79.33**	78.25	79.33	82.33	**78.32**	73.32	**80.32**	**79.07**	59.71	58.82	66.07	56.20	70.46	**70.84**	67.22	59.97	60.00	70.00	28.83	33.83	65.84	60.25
GLM-Image [89]	**81.01**	**81.02**	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BitDance	79.64	78.12	78.79	**80.44**	**83.50**	**87.50**	77.22	**77.62**	75.64	76.21	**72.19**	**71.66**	**72.89**	**78.88**	68.06	66.10	**70.63**	**67.21**	**96.67**	**90.00**	**87.78**	**75.57**	**84.33**	**83.96**

Table 18: **Efficiency comparison of text-to-image generation at 1024×1024 resolution.** Latency is measured using a single H100 GPU with bfloat16 precision.

Model	Type	#Params (B)	Latency-1024 (s)
BAGEL [17]	Diffusion	7	23.1
Qwen-Image [67]	Diffusion	20	20.3
Z-Image [68]	Diffusion	6	21.1
NextStep-1 [66]	Autoregressive	14	402
GLM-Image [89]	Autoregressive	16	53.2
BitDance	Autoregressive	14	12.4

performance gap between open-source and proprietary models while relying on dramatically less training data, underscoring the effectiveness and scalability of our method.

Furthermore, we provide qualitative and quantitative results for both the SFT and adapted models in Fig. 9 and Tab. 17. These results demonstrate that the adapted model maintains excellent generation quality while achieving faster inference speeds. In Tab. 18, we present an efficiency comparison of various text-to-image models generating 1024×1024 images. Compared to both diffusion and autoregressive models, our adapted model achieves faster inference speeds.



Figure 9: **Images generated by SFT and adapted models.**

Table 17: **Quantitative results of the SFT and adapted models.** #Tokens denotes the number of predicted tokens in one decoding step.

Model	#Tokens	DPG-Bench	GenEval
SFT	16	88.28	**0.86**
Adapted	**64**	**88.30**	0.85

Table 19: **Evaluation of image editing** on GEdit-Bench and ImgEdit-Bench. G_SC, G_PQ, and G_O refer to the metrics evaluated by GPT-4.1.

Type	Model	GEdit-Bench-EN (Full set)↑			GEdit-Bench-CN (Full set)↑			ImgEdit-Bench↑
		G_SC	G_PQ	G_O	G_SC	G_PQ	G_O	
Proprietary	Gemini 2.0 [26]	6.73	6.61	6.32	5.43	6.78	5.36	-
	GPT-4o [37]	7.85	7.62	7.53	7.67	7.56	7.30	4.20
Diffusion	Instruct-Pix2Pix [5]	3.58	5.49	3.68	-	-	-	1.88
	MagicBrush [90]	4.68	5.66	4.52	-	-	-	1.83
	AnyEdit [86]	3.18	5.82	3.21	-	-	-	2.45
	OmniGen [77]	5.96	5.89	5.06	-	-	-	2.96
	Step1X-Edit [47]	7.09	6.76	6.70	7.20	6.87	6.86	3.06
	BAGEL [17]	7.36	6.83	6.52	7.34	6.85	6.50	3.42
Autoregressive	NextStep-1 [66]	7.15	7.01	6.58	6.88	7.02	6.40	3.71
	BitDance	7.20	6.78	6.54	7.18	6.82	6.46	3.45

Image Editing. We conduct comprehensive image editing evaluations on GEdit-Bench [47] and ImgEdit-Bench [84], comparing our method with state-of-the-art proprietary models, diffusion-based models, and autoregressive models. Trained with only about 100M data samples, BitDance outperforms Gemini 2.0 [26] and achieves competitive results against leading open-source models, including Step1X-Edit [47], BAGEL [17], and NextStep-1 [66].

D Additional Qualitative Results

We provide additional qualitative results, including ImageNet 256×256 class-conditional image generation (Figs. 14–21) and high-resolution (approximately 1024^2 pixels) text-to-image generation (Figs. 10–13). These results further demonstrate that BitDance is capable of producing high-quality images depicting humans, general objects, and rendered text across diverse styles and aspect ratios.

E Limitations and Broader Impacts

Limitations. Despite the strong empirical results, this work has several limitations. First, our evaluation is mainly focused on class-conditional image generation, text-to-image synthesis, and image editing, and we have not yet validated whether the proposed binary tokenization and next-patch diffusion design generalize to image understanding or unified understanding-generation models. Besides, our current study mainly benchmarks standard datasets and model scales, and broader ablations on tokenizer entropy, patch granularity, and scaling behavior remain for future work.

Broader Impacts. This work contributes to scalable and efficient autoregressive visual generation, which may help broaden access to high-quality image synthesis and editing by reducing the inference cost of AR models. The proposed framework may also provide a useful step toward more unified autoregressive foundation models across vision and language. At the same time, like other powerful generative models, BitDance could be misused to produce misleading, deceptive, or harmful visual content. We therefore encourage future deployment to be accompanied by appropriate safeguards, usage policies, and continued study of fairness, safety, and responsible release practices.



A cinematic film portrait of a beautiful Chinese young woman, with messy black hair blowing in the wind and partially covering her face, her vivid eyes looking directly at the camera. She is positioned in the left third of the frame. She wears a thick bright red knitted scarf and a worn beige shearling coat. The background shows a cold, dry wilderness and distant mountains at sunset. Strong golden backlight shines directly into the lens, creating dramatic lens flare and a hazy glowing halo effect, with a dusty atmosphere in the air. Film grain texture, shallow depth of field, natural and raw style.



A fresh and artistic portrait of a young man. The subject appears clean and natural, with special emphasis on the fine texture and layered detail of the hair. Under backlighting, the hair forms a distinct yet natural glowing-hair effect, with soft and delicate golden rim light along the strands, transitioning smoothly without harsh edges. The background is set at Erhai Lake in Dali, Yunnan, featuring crystal-clear blue water, the hazy rolling outline of the Cangshan Mountains in the distance, and a few small white boats scattered across the lake to enhance depth and a strong sense of place. In the sky, 2 to 4 seagulls of varying shapes are shown in natural flight, with some displaying slight motion blur in their wings to add liveliness and a sense of moving air. A warm golden backlight enters diagonally from behind the subject, either from the upper right or upper left.



A highly atmospheric cinematic portrait photograph of a young woman in a floral dress, sitting alone on a breezy hillside and gazing thoughtfully into the distance. She has long, slightly messy dark wavy hair, and her softly colored floral dress blends naturally with the earthy tones of the surrounding landscape. Thin mist drifts through the mountains, while the light is soft and diffused, creating a scene that feels both dramatic and emotionally rich. Tall dry grasses surround her, swaying gently in the breeze. In the background, a hazy valley gradually disappears into the fog, adding depth and a subtle melancholic mood to the composition. Shot with an 85mm lens, with shallow depth of field to emphasize the subject. Ultra-realistic texture, soft low-saturation tones, moderate contrast, and high-end cinematic color grading.

Figure 10: **Additional text-to-image generation results: human and general objects.**



A stunning long-haired beauty wearing a white fox-fur coat, with a few loose strands of hair tangled in disarray. Mottled light and shadow fall across her face through sparse dead leaves and dense drifting snow dust, creating a fantastical scene. Impressionist style, ultra-wide-angle lens, hazy atmosphere, abstract art, strong artistic mood, high saturation and contrast, masterful lighting, unconventional and unique composition, visual art, award-winning style, with the fur rendered in exquisite detail showing fine snow crystals and frosted luster.



A handsome young man turning his head in a neon-lit street at night, captured with heavy motion blur and soft focus, as if caught in a fleeting cinematic moment. He has medium-length dark hair moving in the wind, sharp facial features, and a calm, melancholic expression. He wears a textured red-orange knit sweater, standing out against a cool cyan and teal urban background filled with streaking lights and dreamy bokeh. Shot on a Leica M6 with Kodak VISION3 500T color negative film, with visible grain, soft halation, tungsten glow, and the emotional atmosphere.



A cinematic film portrait of a beautiful Chinese young woman, with messy black hair blowing in the wind and partially covering her face, her vivid eyes looking directly at the camera. She is positioned in the left third of the frame. She wears a thick bright red knitted scarf and a worn beige shearling coat. The background shows a cold, barren wilderness and distant mountains at sunset. Strong golden backlight shines directly into the lens, creating dramatic lens flare and a hazy glowing halo, with a dusty atmosphere in the air. Film grain texture, shallow depth of field, natural style.



A picturesque still life with pumpkins showcases multiple layers, muted tones, and an expressive, loosely painted composition. The colors are perfectly matched without being flashy, complemented by a gray and light background with linen draperies. Skillful use of color gradation and texture effects creates a vivid world illuminated by light.



In the style of Kim Jung Gi, highly saturated, rich colors, maximalist, asymmetrical composition, perfect detail, masterpiece, storybook style. A red-brown Ferghana horse with a white mane and tail, wearing an ornate saddle. High-detail quality, ultra-high resolution, top quality, clear brushwork, strong light and shadow contrast, cinematic texture.



Dim room facing a centered open window, bright sun through flowers and lush leaves, strong Tyndall rays on a drifting sheer curtain. Cozy cottage corner with a beige lazy sofa, blankets, pillows, bookshelf, oil painting, green plants, and an orange cat sleeping by the window. Healing anime style, Hong Kong aesthetic, warm, cheerful, relaxed.

Figure 11: **Additional text-to-image generation results: human and general objects.**



Within a snow globe, a real-sized rat explores a vibrant miniature LEGO town adorned with charming Christmas decorations. Wreaths, glittering ornaments, and lights beautifully accentuate pink, peach, yellow, and pastel LEGO structures. Whirling white LEGO pieces create the effect of a dynamic snowstorm. Amidst this festive setting, a delicately crafted LEGO sleigh with reindeer embodies simplicity and charm. The composition blends curiosity, celebration, and intricate design into a visually engaging and delightful scene.



A serene silver-haired grandmother, wearing exquisitely detailed medieval armor engraved with intricate patterns and subtle signs of age, is deeply absorbed in reading an ancient, dust-covered book. Her expression is calm and wise, illuminated by soft, atmospheric light that reveals the texture of her weathered skin, flowing silver hair, and finely crafted metal armor. The old book, with worn pages and a timeworn leather cover, suggests forgotten knowledge and long-lost history. Highly realistic, lifelike style, cinematic composition.



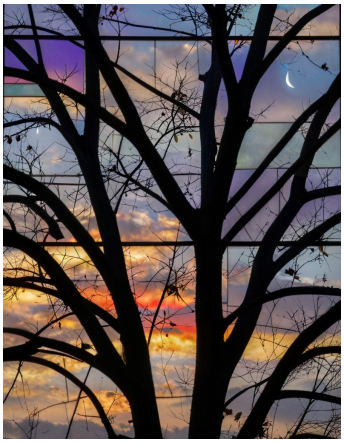
An otherworldly scene of neon-lit jellyfish drifting gracefully through outer space, their translucent bodies glowing in vivid shades of blue, pink, purple, and cyan. Their soft bioluminescent light casts colorful reflections across the surrounding cosmic atmosphere, illuminating swirling nebulae, distant stars, and faint clouds of interstellar dust. The contrast between the delicate floating jellyfish and the vast, mysterious universe creates a mesmerizing, dreamlike tapestry, with a sense of quiet wonder and ethereal beauty.



Create a realistic image of a man carefully loading an antique film reel onto a vintage projector. Warm, vivid light fills the scene, highlighting the texture of aged metal, worn film, and his focused hands. The setting feels nostalgic and authentic, with rich cinematic atmosphere, lifelike detail, and tangible realism. Emphasize natural lighting, subtle shadows, and a timeless mood inspired by classic cinema.



In the heart of an ancient Egyptian temple, torches cast a golden glow over the legendary Mau Cats. Their amber eyes blaze, and their fur shimmers with fiery power, exuding mystical might and wisdom. Sacred flames dance around them, weaving intricate patterns in the air. As guardians of the ancient fire, the Mau Cats stand united, their spirits intertwined with elemental forces. This moment immortalizes them as protectors of ancient wisdom.



Stained-glass style dusk landscape with black leafless tree branches forming intricate leaded outlines across the frame. Fine twigs with a few remaining leaves create layered, airy silhouettes. The sunset sky is divided into vivid glass-like panels of orange, gold, crimson, pale purple, soft blue, and pink-white clouds, with hints of a crescent moon and partial setting sun. blending natural scenery with luminous stained-glass artistry.

Figure 12: **Additional text-to-image generation results: human and general objects.**



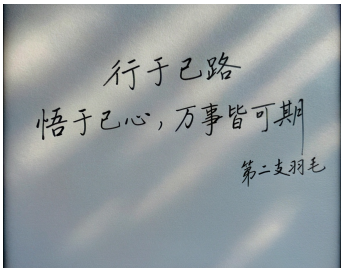
An embroidered-illustration promotional poster themed “阿勒泰印象”. The image features towering pink-and-black mountains capped with snow, dotted with green pine trees, with dense forest at the foot of the hills. On the grassland are white sheep, brown horses, a wooden cabin, and a white Xinjiang-style yurt, with several people near the yurt, depicting the nomadic life of Altay. The title “XinJiang 新疆 阿勒泰印象” is prominent, with “XinJiang” and “新疆” rendered in pink artistic lettering with embroidered texture. The overall palette is soft and poetic, mainly in pink, green, black, and white, conveying both the natural beauty and nomadic charm of Altay. In the foreground, a white sheep and a brown cow stand quietly beside a small wooden house.



Game poster, fantasy RPG key art featuring a glowing sword embedded in ancient stone, set in a misty forest with soft atmospheric light. Cinematic composition, epic yet clean mood, detailed environment, chunky readable typography, minimal UI. Large title: "EMBER QUEST", small line: "NEW SEASON".



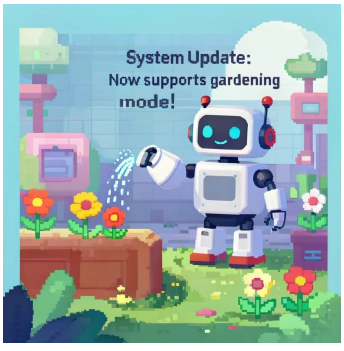
Surreal seascape, an inverted mountain hovering above the ocean, chained slow-falling droplets refracting sunset light, a small boat far away for scale, god rays outlining the mountain edges, realistic material detail. Add large centered text: "UPSIDE DOWN", small text: "HORIZON".



White handwritten artistic signature-style lettering, writing “行于己路 悟于己心，万事皆可期”, with flowing, free-spirited strokes; film-like texture; cool-toned lighting and shadows; a quiet, artistic atmosphere; 8K high definition; ultra-rich details. Signature in the lower right: “第二支羽毛”.



Movie poster, theme: "Endless Road", strong one-point perspective highway into a huge orange-red sun, small car silhouette far away, two bold brush-stroke clouds, retro screen-print texture; add large title: "THE WAY HOME", small text: "A ROAD FILM".



With a clear hovering box displaying "System Update: Now supports gardening mode!", generate pixel art of a cheerful robot watering pixelated flowers in a futuristic garden, with soft neon details, tidy composition, and a playful cozy atmosphere.



A family illustration of a mother decorating a cake as her child watches eagerly at her side, filled with excitement and curiosity. Warm kitchen setting, soft natural light, cozy homemade details, tender emotional atmosphere. Title: "Baking with Love: A Family Story".

Figure 13: **Additional text-to-image generation results: text rendering.**



Figure 14: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "arctic wolf" (270).



Figure 15: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "sulphur-crested cockatoo" (89).



Figure 16: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "cliff drop-off" (972).



Figure 17: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "lion" (291).



Figure 18: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "otter" (360).



Figure 20: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "panda" (388).



Figure 19: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "red panda" (387).



Figure 21: **Uncurated ImageNet 256×256 samples.** BitDance-H-1x. Classifier-free guidance scale = 8.0. Class label = "loggerhead sea turtle" (33).

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly articulate the scope and main contributions of the paper. A concise summary of our contributions is also provided in the introduction.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations of the work in Appendix E.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: This work does not include theoretical derivations.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We provide a clear description of the proposed method in Sec. 3, and detail the experimental setup and implementation in Appendix B and Appendix C.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We use the publicly available ImageNet dataset, and provide code and usage instructions in the Supplementary Material.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Detailed experimental settings are provided in Appendix B and Appendix C.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: Our model is trained on large-scale datasets. Due to limited computational resources, we were unable to conduct extensive statistical significance testing.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We provide details in Appendix B and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: All the authors have reviewed the code of ethics and obey the code.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss the impacts in Appendix E.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: As a purely academic study, this work poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We have cited related original papers in our paper.

    Guidelines:

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

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: Our paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: Our core method development does not involve LLMs.

Guidelines:

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