Infinity∞: Scaling Bitwise AutoRegressive Modeling for High-Resolution Image Synthesis

Jian Han*, Jinlai Liu*, Yi Jiang*, Bin Yan,
Yuqi Zhang, Zehuan Yuan†, Bingyue Peng, Xiaobing Liu
ByteDance

{hanjian.thu123,liujinlai.licio,jiangyi.enjoy,yanbin.master}@bytedance.com
{zhangyuqi.hi,yuanzehuan,bingyue.peng,will.liu}@bytedance.com

Code: https://github.com/FoundationVision/Infinity
Online demo: https://opensource.bytedance.com/gmpt/t2i/invite

Figure 1. High-resolution image synthesis results from our Infinity-2B model, showcasing its capabilities in precise prompt following, spatial reasoning, text rendering, and aesthetics across different styles and aspect ratios.

Abstract

We present Infinity, a Bitwise Visual AutoRegressive Modeling capable of generating high-resolution, photorealistic images following language instruction. Infinity refactors visual autoregressive model under a bitwise token prediction framework with an infinite-vocabulary classifier and bitwise self-correction mechanism. By theoretically expanding the tokenizer vocabulary size to infinity in Transformer, our method significantly unleashes powerful scaling capabilities to infinity compared to vanilla VAR. Extensive experiments indicate Infinity outperforms AutoRegressive Text-to-Image models by large margins, matches or surpasses leading diffusion models. Without extra optimization, Infinity generates a 1024×1024 image in 0.8s, 2.6× faster than SD3-Medium, making it the fastest Text-to-Image model. All the code and models are available to promote further exploration of Infinity for visual generation.

*Equal contribution. †Corresponding author.

1. Introduction

Visual generation[20, 27, 45, 51, 55] has recently seen rapid advancements, enabling high-quality, high-resolution images and video synthesis[7, 22]. Text-to-image generation[6, 22, 46, 48, 49, 53] is one of the most challenging tasks due to its need for complex language adherence and intricate scene creation. Currently, visual generation is primarily divided into two main approaches: Diffusion models and AutoRegressive models.

Diffusion models[20, 22, 27, 45, 46, 55], trained to invert the forward paths of data toward random noise, effectively generate images through a continuous denoising process. AutoRegressive models[14, 21, 21, 58, 64, 73], on the other hand, harness the scalability and generalizability of language models[2, 3, 16, 28, 60, 63, 65, 66, 72] by employing a visual tokenizer[50, 67, 78] to convert images into discrete tokens and causally optimize these tokens, allowing image generation through next-token prediction or next-scale prediction. AutoRegressive models encounter significant challenges in high-resolution text-to-image synthesis[58, 70, 79]. They exhibit inferior reconstruction quality when utilizing discrete tokens as opposed to continuous tokens. Additionally, their generated visual contents are less detailed than those by diffusion models. Inefficiency and latency in visual generation, resulting from the raster scan method of next-token prediction, further exacerbate these issues.

Recently, Visual AutoRegressive Modeling (VAR)[64] redefined autoregressive learning on images as coarse-to-fine "next-scale prediction". It demonstrates superior generalization and scaling capabilities compared to diffusion transformers while requiring fewer steps. VAR leverages the powerful scaling properties of LLMs [25, 31] and can simultaneously refine previous scale steps, benefiting from the strengths of diffusion models as well. However, the index-wise discrete tokenizer[21, 38, 42, 64, 67, 69, 83] employed in AutoRegressive or Visual AutoRegressive models faces significant quantization errors with a limited vocabulary size resulting in difficulties in reconstructing fine-grained details, especially in high-resolution images. In the generation stage, index-wise tokens suffer from fuzzy supervision leading to visual detail loss and local distortions. Moreover, train-test discrepancies from teacher-forcing training, inherent to LLMs, amplify cumulative errors in visual details. These challenges make index-wise tokens a significant bottleneck for AutoRegressive models.

We propose a novel bitwise modeling framework that substitutes index-wise tokens with bitwise tokens throughout the process. Our bitwise modeling framework consists of three primary modules: bitwise visual tokenizer, bitwise infinite-vocabulary classifier, and bitwise self-correction. Inspired by the success and widespread adoption of binary vector quantization[80, 85], we scaled up the tokenizer vocabulary to 2^{64}, significantly surpassing all previous AutoRegressive model vocabularies[58, 81]. This expansion allows for reconstruction quality that far exceeds previous discrete tokenizers, achieving results comparable to continuous VAE[51], with scores improving from 0.87 to 0.33 on ImageNet-256 benchmark[19]. In Fig.2, we transform the conventional token prediction from a large integer into binary bits in a bitwise infinite-vocabulary classifier to address optimization and computation challenges, enabling the learning of massive vocabularies in Visual AutoRegressive models. Additionally, we incorporated bitwise self-correction during training by randomly flipping some bits to simulate prediction mistakes and re-quantizing the residual features, thus endowing the system with self-correcting capabilities. Our method, Infinity: Bitwise Visual AutoRegressive Modeling, maintains the scaling and speed advantages of VAR while achieving detailed reconstruction and generation quality comparable to that of continuous Diffusion models.

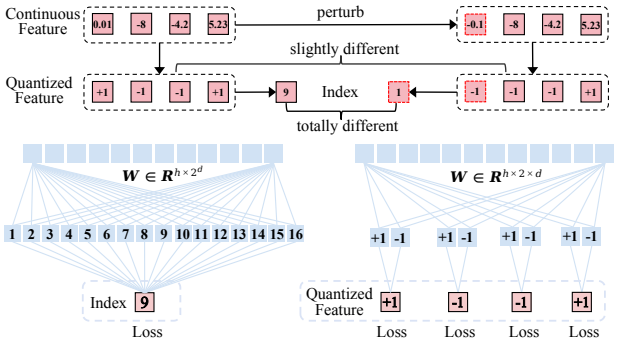

Figure 2. Left: Conventional classifier predicts 2^d indices. Right: Infinite-Vocabulary Classifier predicts d bits instead, being much more efficient than index-wise classifier when d is large.

Infinity outperforms all previous AutoRegressive text-to-image models by large margins and matches or surpasses state-of-the-art diffusion models including SDXL[46], PixArt-Sigma[13],DALL-E3[6] and Stable-Diffusion 3[22] in both quantitative evaluations and human preference scores. The core contributions of our work are as follows:

1. We propose Infinity, an autoregressive model utilizing Bitwise Modeling, which significantly enhances scalability and visual detail representation capabilities of discrete generative models. This framework opens up new possibilities of 'infinity' within the discrete generation community.
2. Infinity demonstrates the potential of scaling tokenizers and transformers by achieving near-continuous tokenizer performance with its image tokenizer, surpassing diffusion models in high-quality text-to-image generation.
3. Infinity enables an autoregressive text-to-image model to achieve exceptional prompt adherence and superior

image generation quality, while also delivering unprecedented inference speed.

2. Related Work

2.1. AutoRegressive Models

AutoRegressive models, leveraging the powerful scaling capabilities of LLMs[8, 16, 47, 65, 66], use discrete image tokenizers[21, 50, 67] in conjunction with transformers to generate images based on next-token prediction. VQ-based methods [21, 35, 50, 58, 67] employ vector quantization to convert image patches into index-wise tokens and use a decoder-only transformer to predict the next token index. However, these methods are limited by the lack of scaled-up transformers and the quantization error inherent in VQ-VAE[67], preventing them from achieving performance on par with diffusion models. Parti [79], Emu3 [70], chameleon[62] , Liquid[73], and VideoPoet[32] scaled up autoregressive models in text-to-image or video synthesis. Inspired by the spatial structure of visual information, Visual AutoRegressive modeling(VAR)[64] redefines the autoregressive modeling on images as next-scale prediction framework, significantly improving generation quality and sampling speed. HART[61] adopted hybrid tokenizers based on VAR. MAR[37], Fluid[23] proposed random-order models and employed continuous tokenizer rather than discrete tokenizer.

2.2. Diffusion Models.

Diffusion models have recently experienced rapid advancements in several directions. Denoising learning mechanisms [27, 44] and improvements in sampling efficiency [4, 40, 41, 55, 56] have been continuously optimized to generate high-quality images. Bit Diffusion [15] represents data with analog bits and then trains a continuous diffusion model. Latent diffusion models [51] were the first to propose modeling in the latent space rather than the pixel space. Recently, latent diffusion models [18, 22] have scaled up VAEs to enhance representations in the latent space. DiT [45] and U-ViT [5] employ more scalable transformers to model diffusion, achieving superior results. Consequently, mainstream text-to-image diffusion models [6, 12, 22] have adopted the DiT architecture due to its effectiveness.

2.3. Scaling Law

Scaling laws in autoregressive language models reveal a power-law relationship between model size, dataset size, and compute with test set cross-entropy loss [1, 25, 31]. These laws help predict larger model performance, leading to efficient resource allocation and ongoing improvements without saturation [8, 28, 65, 66, 72, 84]. This has inspired research into scaling in visual generation [7, 22, 59, 64, 82].

3. Infinity Setup

3.1. Visual AutoRegressive Modeling

Infinity incorporates a visual tokenizer and a transformer for image synthesis. During the training stage, a sample consists of a text prompt t and a ground truth image im. The proposed visual tokenizer first encodes the image im into a feature map $f \in \mathbb{R}^{h\times w\times C}$ with stride s and then quantizes the feature map f into K multi-scale residual maps $(r_1, r_2, ..., r_K)$. The resolution of r_k is $h_k \times w_k$ and it grows larger gradually from $k = 1 \rightarrow K$. Based on this sequence of residuals, we can gradually approximate the continuous feature f as in Eq.1

$$f_k = \sum_{i=1}^{k} up(r_i, (h, w)) \tag{1}$$

up means bilinear upsampling and f_k is the cumulative sum of the upsampled $r_{\leq k}$.

Subsequently, transformer learns to predict residuals r of the next scale conditioned on previous predictions and the text input in an autoregressive manner. Formally, the autoregressive likelihood can be formulated as:

$$p(r_1, ..., r_K) = \prod_{k=1}^{K} p(r_k \mid r_1, ..., r_{k-1}, \psi(t)), \tag{2}$$

where $\psi(t)$ is the text embeddings from Flan-T5 [17] model. $(r_1, ..., r_{k-1}, \psi(t))$ serves as the prefixed context When predicting r_k. Besides, the text embeddings $\psi(t)$ further guide the prediction through a cross-attention mechanism. In particular, as shown in Fig. 3, the text embeddings $\psi(t) \in \mathbb{R}^{L\times C}$ is projected into a $\langle \text{SOS} \rangle \in \mathbb{R}^{1\times 1\times h}$ as the input of the first scale, where h is the hidden dimension of transformer. The transformer is required to predict r_1 based on $\langle \text{SOS} \rangle$ in the first scale. In the latter k-th scale, to match the spatial size of the input and the output label R_k, we take the downsampled feature $\widetilde{f}_{k-1}$ from the last scale $k-1$ as the input to predict R_k in parallel.

$$\widetilde{f}_{k-1} = down(f_{k-1}, (h_k, w_k)), \tag{3}$$

where *down* is bilinear downsampling and the spatial size of both $\widetilde{f}_{k-1}$ and r_k is (h_k, w_k). In previous index-wise [64] representations, the shape of r_k is (h_k, w_k, V_d). V_d is the vocabulary size of the visual tokenizer. For binary quantization [80, 85] with code embedding dimension d, $V_d = 2^d$.

The transformer consists of a stack of repeated blocks, where each block includes RoPE2D [26], Self-Attention, Cross Attention, and FFN layers. The text embeddings $\psi(t)$ provide effective guidance for image synthesis in each cross-attention layer. During the training stage, we exploit a block-wise causal attention mask to ensure that the

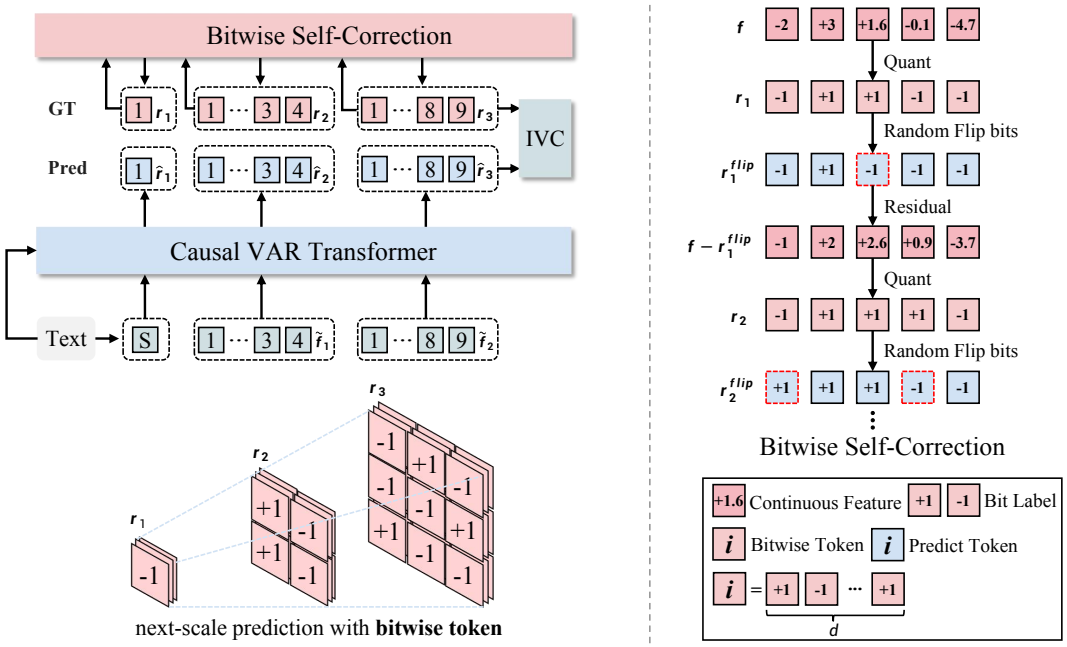

Figure 3. **Framework of Infinity.** Infinity introduces bitwise modeling. When predicting r_k, the sequence $(r_1, r_2, ..., r_{k-1})$ serves as the prefixed context and the text embeddings $\psi(t)$ guides the prediction through a cross attention mechanism.

transformer can only attend to its prefixed context, *i.e.*, $(\langle \text{SOS} \rangle, \widetilde{f}_1, ..., \widetilde{f}_{k-1})$, when predicting f_k. During the inference stage, we perform KV-Caching to speed up inference and there's no need for masking.

3.2. Infinite Tokenizer

Increasing the vocabulary size has significant potential for improving reconstruction and generation quality. However, directly enlarging the vocabulary in existing tokenizers[58, 64, 71, 81] leads to a substantial increase in memory consumption and computational burden. To address these challenges and fully exploit the potential of discrete tokenizers, this paper proposes a new **bitwise multi-scale residual quantizer**, which significantly reduces memory usage, enabling the training of extremely large vocabulary, e.g. 2^{64}.

Bitwise Multi-scale Residual Quantizer. We replace the original vector quantizer of VAR [64] with a dimension-independent bitwise quantizer. In this paper, we consider two candidates, LFQ [81] and BSQ[85]. Given K scales in the multi-scale quantizer, on the k-th scale, the input continuous residual vector $z_k \in \mathbb{R}^d$ are quantized into binary output q_k as shown below.

$$q_k = \begin{cases} sign(z_k) & \textit{lookup-free quantizer} \\ \frac{1}{\sqrt{d}} sign(\frac{z_k}{|z_k|}) & \textit{binary spherical quantizer} \end{cases} \tag{4}$$

To encourage codebook utilization, an entropy penalty $\mathcal{L}_{entropy} = \mathbb{E}[H(q(z))] - H[\mathbb{E}(q(z))]$ [30] is adopted. To obtain the distribution of $q(z)$, we need to compute the similarities between the input z and the whole codebook when using LFQ. However, this leads to unaffordable space and time complexity of $O(2^d)$. When the codebook dimension d becomes large, e.g. 20, an out-of-memory (OOM) issue is faced as shown in Tab. 3. By contrast, since both input and output in BSQ are unit vectors, BSQ[85] proposes an approximation formula for the entropy penalty, reducing the computational complexity to $O(d)$. As shown in Tab 3, there is no obvious increase in memory consumption for BSQ even when codebook size is 2^{64}. Unless otherwise stated, we adopt BSQ by default.

3.3. Infinite-Vocabulary Classifier

The visual tokenizer obtains discrete labels by quantizing residual features. Consequently, the transformer predicts next-scale residual features' labels $\mathbf{y}_k \in [0, V_d)^{h_k \times w_k}$ and optimizes the target through the cross-entropy loss. Previous works [64, 80] directly predict these index labels using a classifier of V_d classes. However, it suffers from two drawbacks, huge computational costs and fuzzy supervision.

As illustrated in Section 3.2, we exploit a bitwise VQ-VAE as the visual tokenizer, where the vocabulary size V_d is extremely large. For example, if $V_d = 2^{32}$ and $h = 2048$, a conventional classifier would require a weight matrix $W \in \mathbb{R}^{h \times V_d}$ of 8.8 trillion parameters, which exceeds the limits of current computational resources.

Moreover, VQ-VAE exploits the sign function during quantization as in Eq.4. After that, the positive elements are multiplied with the corresponding base and summed to get the index label $\mathbf{y}_k(m, n)$ as in Eq.5, where $m \in [0, h_k)$ and $n \in [0, w_k)$.

$$\mathbf{y}_k(m,n) = \sum_{p=0}^{d-1} \mathbb{1}_{r_k(m,n,p)>0} \cdot 2^p \tag{5}$$

Owing to the merits of the quantization method, slight perturbations to those near-zero features cause a significant change in the label. As a result, the conventional index-wise classifier [10, 64, 81] is difficult to optimize.

To address these problems in computation and optimization, we propose an Infinite-Vocabulary Classifier (IVC). In particular, instead of using a conventional classifier with V_d classes, we use d binary classifiers in parallel to predict if the next-scale residual $r_k(m,n,p)$ is positive or negative, where $d = log_2(V_d)$. The proposed Infinite-Vocabulary Classifier is much more efficient in memory and parameters compared to the conventional classifier. When $V_d = 2^{16}$ and $h = 2048$, it saves 99.95% parameters and GPU memory. Besides, when there exist near-zero values that confuse the model in some dimensions, the supervision in other dimensions is still clear. Therefore, compared with conventional index-wise classifiers, the proposed Infinite-Vocabulary Classifier is easier to optimize.

3.4. Bitwise Self-Correction

Weakness of teacher-forcing training. VAR [64] inherits the teacher-forcing training from LLMs. However, next-scale prediction in vision is quite different from next-token prediction in language. Specifically, we cannot decode the complete image until residuals r_k from all scales are obtained. We find that the teacher-forcing training brings about severe train-test discrepancies for visual generation. In particular, the teacher-forcing training makes the transformer only refine features in each scale without the ability to recognize and correct mistakes. Mistakes made in former scales will be propagated and amplified in latter scales, finally messing up generated images (left images in Fig.10).

In this work, we propose the bitwise self-correction to address this issue. In particular, we obtain r_k^{flip} via randomly flipping the bits in r_k with a probability from 0% to 30%, imitating the errors in the prediction of the k-th scale.

Here comes the key component of bitwise self-correction. r_k^{flip} contains errors while r_k doesn't. After replacing r_k with r_k^{flip} as predictions on the k-th scale, we need to recompute the transformer input $\widetilde{f}_k$. Besides, re-quantization is performed to get a new target r_{k+1} as in Eq.6. Refer to Fig.3 (right) for a simplified illustration.

$$\begin{cases} f_k^{flip} = \sum_{i=1}^{k} up(r_i^{flip}, (h,w)) \\ \widetilde{f}_k = down(f_k^{flip}, (h_{k+1}, w_{k+1})) \\ r_{k+1} = quant(down(f - f_k^{flip}, (h_{k+1}, w_{k+1}))) \end{cases} \tag{6}$$

Each scale undergoes the same process of random-flipping and re-quantization. Here the transformer takes partially randomly flipped features as inputs, taking the prediction errors into consideration. The re-quantized bit labels enable the transformer to auto-correct errors made in former predictions. In such a way, we address the train-test discrepancy caused by teacher-forcing training and empower Infinity to have the self-correction ability.

3.5. Dynamic Resolution Generation

Infinity can generate photo-realistic images with various aspect ratios, which is significantly different from VAR [64] that can only generate square images. The main obstacles of generating various aspect ratio images lie in two folds. The first is to define the height h_k and width w_k of r_k based on varying aspect ratios. In the supplementary material, we pre-define a list of scales, also called scale schedule, as $\{(h_1^r, w_1^r), ..., (h_K^r, w_K^r)\}$ for each aspect ratio. We ensure that the aspect ratio of each tuple (h_k^r, w_k^r) is approximately equal to r, especially in the latter prediction scales. Additionally, for different aspect ratios at the same scale k, we keep the area of $h_k^r \times w_k^r$ to be roughly equal, ensuring that the training sequence lengths are roughly the same.

Secondly, we need to carefully design a resolution-aware positional encoding method to handle features of various scales and aspect ratios. This issue poses a significant challenge, as the existing solutions [26, 43, 57, 64, 68] exhibit substantial limitations under such conditions. In this paper, we apply RoPE2d [26] on features of each scale to preserve the intrinsic 2D structure of images. Additionally, we exploit learnable scale embeddings to avoid confusion between features of different scales. Compared to learnable APE element-wise applied on features, learnable embeddings applied on scales bring fewer parameters, can adapt to varying sequence lengths, and are easier to optimize.

4. Experiment

4.1. Dataset

Data Curation. We curated a large-scale dataset from open-source academic data and high-quality internally collected data. The pre-training dataset is constructed by collecting and cleaning open-source academic datasets such as LAION [54], COYO [9], OpenImages [33]. We exploit an OCR model and a watermark detection model to filter undesired images with too many texts or watermarks. Additionally, we employ Aesthetic-V2 to filter out images with low aesthetic scores.

4.2. Training

Progressive Training. Infinity redefines text-to-image as a coarse-to-fine, next-scale prediction task. In line with its architecture, we propose to train Infinity in a progressive

strategy. Specifically, we first train Infinity of 2B parameters on the pre-training dataset of over 100 million images with 256 resolution for 150k iterations using a batch size of 4096 and a learning rate of 6e-5. Then we switch to 512 resolution and train 110k iterations using the same hyper-parameters. Next, we fine-tune Infinity at 1024 resolution with a smaller, high-quality dataset of 60 million images. In this stage, we train Infinity for 60k iterations using a batch size of 2048 and a learning rate of 2e-5. All training stages use images of varying aspect ratios.

4.3. Text-to-Image Generation

4.3.1 Qualitative Results

Overall Results. Fig.1 presents generated images from our Infinity-2B model, showcasing Infinity's strong capabilities in generating high-fidelity images from various categories following user prompts. Qualitative comparison results among Infinity and other top-tier models can be found in the supplementary materials.

Prompt-Following. Fig.11 in the appendix presents three examples demonstrating the superior prompt-following ability of Infinity. As highlighted in red, Infinity consistently adheres to user prompts, whether they are short or extremely long. We attribute these improvements to scaling autoregressive modeling and visual tokenizer's vocabulary.

Text Rendering. As illustrated in Fig.12 in the appendix, Infinity can render text according to user prompts across diverse categories. Despite diverse backgrounds and subjects, Infinity accurately renders corresponding texts according to user requirements, such as fonts, styles, colors, and more.

4.3.2 Quantitative Results

Benchmark. As in Tab 1, on GenEval[24], our model with a re-writer achieves the best overall score of 0.73. Besides, Infinity also reaches the highest position reasoning score of 0.49. On DPG [29]. Our model reaches an overall score of 83.5, surpassing SDXL [46], Playground v2.5 [36], and DALLE 3 [6]. What's more, Infinity achieves the best relation score of 90.76 among all open-source T2I models, demonstrating its stronger ability to generate spatially consistent images based on user prompts.

Human Preference Evaluation. We conduct human preference evaluation in both human studies and benchmarks. As in Fig.5, the generation results of Infinity are more frequently selected by humans in terms of *overall quality, prompt following, and visual aesthetics* in contrast to other open-sourced T2I models. Please refer to the supplementary material for more details. Tab.2 lists the results of three human preference benchmarks, *i.e.*, ImageReward [77], HPSv2 [74], and VQAScore[39]. Infinity reaches the highest ImageReward and HPSv2.1, indicating our method could generate images that are more appealing to humans.

Methods	# Params	GenEval↑				DPG↑		
		Two Obj.	Position	Color Attri.	**Overall**	Global	Relation	**Overall**
Diffusion Models								
LDM [52]	1.4B	0.29	0.02	0.05	0.37	-	-	-
SDv1.5 [52]	0.9B	0.38	0.04	0.06	0.43	74.63	73.49	63.2
PixArt-alpha [11]	0.6B	0.50	0.08	0.07	0.48	74.97	82.57	71.1
SDv2.1 [52]	0.9B	0.51	0.07	0.17	0.50	77.67	80.72	68.1
DALL-E 2 [49]	6.5B	0.66	0.10	0.19	0.52	-	-	-
DALL-E 3 [6]	-	-	-	-	0.67†	90.97	90.58	83.5
SDXL [46]	2.6B	0.74	0.15	0.23	0.55	83.27	86.76	74.7
PixArt-Sigma [13]	0.6B	0.62	0.14	0.27	0.55	86.89	86.59	80.5
SD3 (d=24) [22]	2B	0.74	0.34	0.36	0.62	-	-	84.1
SD3 (d=38) [22]	8B	0.89	0.34	0.47	0.71	-	-	-
SANA-1.0 [75]	1.6B	-	-	-	0.66	-	-	84.8
FLUX-dev [34]	12B	-	-	-	0.67	-	-	84.0
FLUX-schnell [34]	12B	-	-	-	0.71	-	-	84.8
AutoRegressive Models								
LlamaGen [58]	0.8B	0.34	0.07	0.04	0.32			65.2
Chameleon [62]	7B	-	-	-	0.39	-	-	-
HART [61]	0.7B	-	-	-	0.56	-	-	80.9
Show-o [76]	1.3B	0.80	0.31	0.50	0.68	-	-	67.5
Emu3 [70]	8.5B	0.81†	0.49†	0.45†	0.66†	-	-	81.6
Infinity	2B	0.85†	**0.49**†	**0.57**†	**0.73**†	**93.11**	**90.76**	83.5

Table 1. Evaluation on the GenEval [24] and DPG [29] benchmark. A blue background indicates methods with larger model sizes than ours. † result is with prompt rewriting.

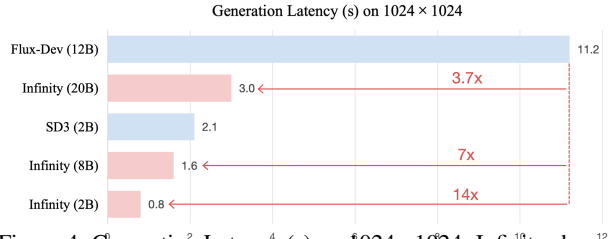

Figure 4. Generation Latency (s) on 1024×1024. Infinity shows a significant advantage in generation latency compared to diffusion models.

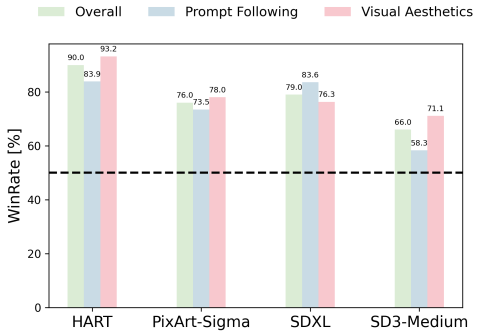

Figure 5. Human Preference Evaluation. We ask users to select the better one in a side-by-side comparison in terms of Overall Quality, Prompt Following, and Visual Aesthetics. Infinity is more preferred by humans compared to other methods.

Inference Latency. As shown in Fig.4, Infinity demonstrates a significant advantage in generation speed compared to diffusion models across different model sizes. Without extra optimization, Infinity generates a 1024×1024 image in 0.8s, 2.6× faster than SD3-Medium. The speed advantage becomes more substantial as the model size increases. Infinity achieves 7× faster inference latency compared to SD3.5 (8B) [22] and Flux-Dev (12B). Besides, Infinity still achieves 3.7× speedup than Flux-Dev even with more parameters (20B v.s. 12B).

Methods	# Params	ImageReward↑		HPSv2.1↑		VQAScore↑	
		Rank	Score	Rank	Score	Rank	Time
SD-XL [46]	2.6B	4	0.600	4	30.06	3	0.832
SD3-Medium [22]	2B	3	0.871	3	30.91	1	**0.906**
PixArt Sigma [13]	0.6B	2	0.872	2	31.47	-	-
Infinity	2B	1	**0.953**	1	**32.05**	2	0.894

Table 2. **Human Preference Metrics**. Infinity reaches the highest ImageReward and HPSv2.1 score, also with comparable VQAScore to SD3.

4.4. Scaling Visual Tokenizer's Vocabulary

Scaling up the Vocabulary Benefits Reconstruction. Restricted by the vocabulary size, discrete VQ-VAEs have always lagged behind continuous ones, hindering the performance of AR-based T2I models. In this work, we successfully train a discrete VQ-VAE matching its continuous counterparts by scaling up the vocabulary size. As in Tab. 4, we observe consistent rFID improvements as scaling up the vocabulary size from 2^{16} to 2^{64}. It's noteworthy that our discrete tokenizer achieves a rFID of 0.61 on ImageNet-256 when $V_d = 2^{32}$, outperforming SD [52]'s contiguous VAE.

Infinite Vocabulary Classifier Benefits Generation. We compare predicting bit labels with IVC to predicting index labels using a conventional classifier under the vocabulary size of 2^{16} since a larger vocabulary causes OOM for the conventional classifier. We utilize the reconstruction loss on $\mathbf{R}_k$, the Fréchet Inception Distance (FID) on the validation dataset and ImageReward for comprehensive evaluation. As shown in Tab.5, IVC achieves lower reconstruction loss and FID, suggesting IVC has better fitting capabilities. Beyond the quantitative results, training Infinity with IVC yields images with richer details as in Fig.6, which is consistent with a higher ImageReward.

Scaling Up the Vocabulary Benefits Generation. We then scale up the vocabulary size to 2^{32} during training the T2I model, which exceeds the range of the Int32 data type and can be considered infinitely large. In Fig.7, we illustrate the effect of scaling up the vocabulary from 2^{16} to 2^{32} for image generation. For small models (125M and 361M), the vocabulary size of 2^{16} converges faster and achieves better results. However, as we scaled up the transformer to 2.2B, the vocabulary size of 2^{32} beats 2^{16} after 40K iterations. Therefore, it's worthwhile to scale up the vocabulary along with scaling up the transformer. As illustrated in Tab.1,2, with infinite vocabulary and IVC, Infinity achieves superior performance among various benchmarks, redefining performance limits in autoregressive visual synthesis.

4.5. Scaling Visual AutoRegressive Modeling

In Fig.8, we depict the validation loss against the total training iterations and computational FLOPs for various model sizes of Infinity. We consistently notice a reduction in validation loss with an increase in training steps and computational FLOPs. Nevertheless, the advantages gained from training smaller models for extended periods lag behind those obtained from training larger models for shorter durations. This trend aligns with findings in language models, emphasizing the promising outlook for increasing model sizes with appropriate training.

In Fig.8, we plot GenEval, ImageReward, and HPSv2 scores against validation loss for different model sizes ranging from 125M to 4.7B. We observe a strong correlation between validation loss and evaluation metrics. To further quantify their correlation, we calculate the Pearson correlation coefficients through linear regression. The correlation coefficients for GenEval, ImageReward, and HPSv2 are -0.983, -0.981, and -0.979, respectively. These results demonstrate a nearly linear correlation between validation loss and the evaluation metrics when scaling up model sizes from 125M to 4.7B. This promising phenomenon encourages us to scale up Infinity to achieve better performance.

Quantizer	$d = 16$	$d = 18$	$d = 20$	$d = 32$	$d = 64$
LFQ	37.6	53.7	OOM	OOM	OOM
BSQ	32.4	32.4	32.4	32.4	32.4

Table 3. Comparison of memory consumption (GB) between different quantizers during training. As codebook dimension d increases, MSR-BSQ shows significant advantages over MSR-LFQ, enabling nearly infinite vocabulary size of 2^{64}.

VAE (stride=16)	TYPE	IN-256 rFID↓	IN-512 rFID↓
$V_d = 2^{16}$	Discrete	1.22	0.31
$V_d = 2^{24}$	Discrete	0.75	0.30
$V_d = 2^{32}$	Discrete	0.61	0.23
$V_d = 2^{64}$	Discrete	**0.33**	**0.15**
SD VAE [52]	Continual	0.87	N/A

Table 4. By scaling up visual tokenizer's vocabulary, discrete tokenizer surpasses continuous VAE of SD [51] on ImageNet-rFID.

Classifier	# Params	vRAM	Recons. Loss↓	FID↓	ImageReward↑
Convention	124M	2GB	0.184	5.95	0.79
IVC	0.65M	10MB	0.180	5.32	0.91

Table 5. In contrast to the conventional classifier, IVC saves 99.95% parameters (vRAM) and reaches better performance.

Figure 6. **Impact of Infinite-Vocabulary Classifier**. Predicting bitwise labels with the Infinite-Vocabulary Classifier (Right) generates images with richer details compared to predicting index-wise labels using a conventional classifier (Left).

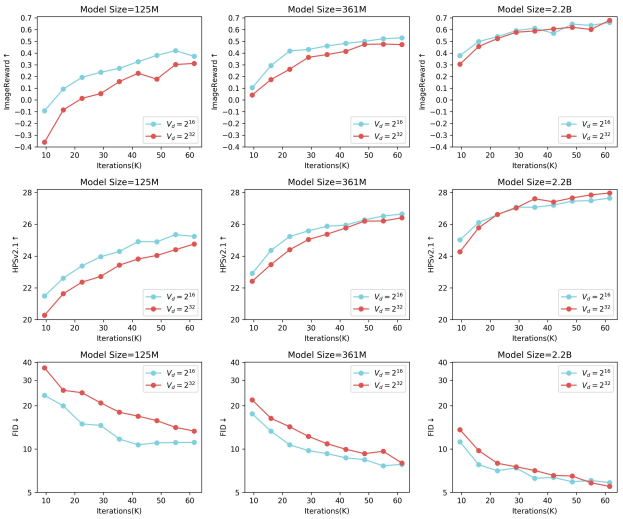

Figure 7. **Effects of Scaling Up the Vocabulary.** We analyze the impact of scaling the vocabulary size under consistent training hyperparameters. Vocabulary size $V_d = 2^{16}$ converges faster and achieves better results for small models (125M and 361M parameters). As we scale up the model size to 2.2B, Infinity with a vocabulary size $V_d = 2^{32}$ beats $V_d = 2^{16}$. Experiment with 5M high-quality data under 256×256 resolution.

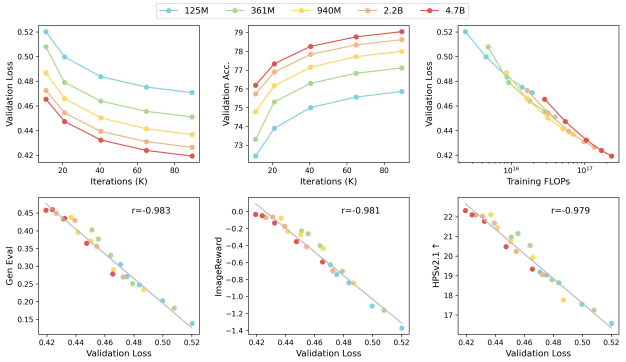

Figure 8. **Scaling Visual AutoRegressive Modeling effects.** We analyze the impact of scaling model size under consistent training hyperparameters (Experiment with 10M pre-training data under 256 resolution). Validation loss smoothly decreases as a function of the model size and training iters. Besides, validation loss is a strong predictor of overall performance. There is a strong correlation between validation loss and holistic image evaluation metrics.

Visualization of scaling effect. To delve deeper into the scaling effect of Infinity, we compare a set of generated 256×256 images of three model sizes (125M, 940M, 4.7B) across three distinct training schedules (10K, 40K, 90K iterations) as illustrated in Fig.9. The semantics and visual quality of generated images improve steadily when scaling up model size and training compute, which is consistent with the scaling behaviors of Infinity.

4.6. Bitwise Self-Correction

In Tab.6 and Fig.10, we list the evaluation metrics and present images generated by models trained using teacher-forcing and bitwise self-correction methods. Substantial advantages are observed after applying bitwise self-correction, primarily driven by the self-correction mechanism rather than applying flipping. Simply random flipping $\mathbf{R}_k$ doesn't bring improvements. Self-correction considers errors and performs re-quantization to correct them. We emphasize that self-correction is essential for AR-based T2I models since it empowers models to correct errors automatically, significantly mitigating the train-test discrepancy.

Figure 9. Semantics and visual quality improve consistently with scaling up model size and training compute.

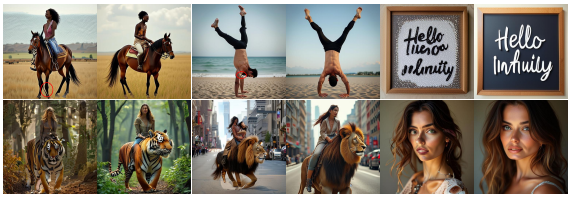

Figure 10. **Impact of Self-Correction**. Teacher-forcing training introduces great train-test discrepancy which degrades performance during inference (left). Bitwise Self-Correction autocorrects mistakes and thus generates better results (right).

Method	FID↓	ImageReward↑	HPSv2.1↑
Baseline	9.76	0.52	29.53
Baseline + Random Flip	9.69	0.52	29.20
Baseline + Bitwise Self-Correction	**3.47**	**0.76**	**30.71**

Table 6. Bitwise Self-Correction makes significant improvements. Experiment with 5M high quality data and 512×512 resolution. FID is measured on the validation set with 40K images.

5. Conclusion

We introduce Infinity, a bitwise visual autoregressive model to perform Text-to-Image generation. Infinity is a pioneering framework for bitwise token modeling with the IVC and self-correction innovation. Extensive results demonstrate Infinity significantly raised the upper limit for Autogressive Text-To-Image models, matching or surpassing leading diffusion models. We believe Infinity will significantly promote the development of autoregressive visual modeling and inspire the community for faster and more realistic generation models.

6. Acknowledgment

Many colleagues from ByteDance supported this work. We are grateful to Guanyang Deng for his efforts in data processing. We also thank Chongxi Wang and Taekmin Kim for their contributions to model deployment. Special thanks to Xiaoxiao Qin for her work in human preference evaluation. Additionally, we are thankful to Hui Wu, Fu Li, Xing Wang, Hongxiang Hao, and Chuan Li for their contributions to infrastructure.

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. 3

[2] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. 2

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2

[4] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022. 3

[5] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22669–22679, 2023. 3

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

[7] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. *OpenAI*, 2024. 2, 3

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

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

[10] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022. 5

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

[12] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 3

[13] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-sigma: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024. 2, 6, 7

[14] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pages 1691–1703. PMLR, 2020. 2

[15] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022. 3

[16] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023. 2, 3

[17] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024. 3

[18] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 3

[19] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[20] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[21] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 2, 3

[22] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 2, 3, 6, 7

[23] 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, 2024. 3

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

[25] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020. 2, 3

[26] Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pages 289–305. Springer, 2025. 3, 5

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3

[28] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022. 2, 3

[29] 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. 6

[30] Aren Jansen, Daniel PW Ellis, Shawn Hershey, R Channing Moore, Manoj Plakal, Ashok C Popat, and Rif A Saurous. Coincidence, categorization, and consolidation: Learning to recognize sounds with minimal supervision. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 121–125. IEEE, 2020. 4

[31] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 2, 3

[32] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Josh Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan Yang, Irfan Essa, Huisheng Wang, David A. Ross, Bryan Seybold, and Lu Jiang. Videopoet: A large language model for zero-shot video generation, 2024. 3

[33] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020. 5

[34] Black Forest Labs. Flux. https://blackforestlabs.ai/announcing-black-forest-labs/, 2024. 6

[35] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532, 2022. 3

[36] Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation. *arXiv preprint arXiv:2402.17245*, 2024. 6

[37] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024. 3

[38] Xiang Li, Hao Chen, Kai Qiu, Jason Kuen, Jiuxiang Gu, Bhiksha Raj, and Zhe Lin. Imagefolder: Autoregressive image generation with folded tokens. *arXiv preprint arXiv:2410.01756*, 2024. 2

[39] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision*, pages 366–384. Springer, 2024. 6

[40] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 3

[41] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 3

[42] Chuofan Ma, Yi Jiang, Junfeng Wu, Jihan Yang, Xin Yu, Zehuan Yuan, Bingyue Peng, and Xiaojuan Qi. Unitok: A unified tokenizer for visual generation and understanding. *arXiv preprint arXiv:2502.20321*, 2025. 2

[43] Xiaoxiao Ma, Mohan Zhou, Tao Liang, Yalong Bai, Tiejun Zhao, Huaian Chen, and Yi Jin. Star: Scale-wise text-to-image generation via auto-regressive representations. *arXiv preprint arXiv:2406.10797*, 2024. 5

[44] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021. 3

[45] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 2, 3

[46] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2, 6, 7

[47] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 3

[48] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

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

[50] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019. 2, 3

[51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3, 7

[52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 6, 7

[53] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 2

[54] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 5

[55] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3

[56] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 3

[57] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 5

[58] 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, 4, 6

[59] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*, 2023. 3

[60] Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, et al. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv preprint arXiv:2107.02137*, 2021. 2

[61] Haotian Tang, Yecheng Wu, Shang Yang, Enze Xie, Junsong Chen, Junyu Chen, Zhuoyang Zhang, Han Cai, Yao Lu, and Song Han. Hart: Efficient visual generation with hybrid autoregressive transformer. *arXiv preprint arXiv:2410.10812*, 2024. 3, 6

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

[63] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 2

[64] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *arXiv preprint arXiv:2404.02905*, 2024. 2, 3, 4, 5

[65] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2, 3

[66] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 2, 3

[67] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 2, 3

[68] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 5

[69] Junke Wang, Yi Jiang, Zehuan Yuan, Bingyue Peng, Zuxuan Wu, and Yu-Gang Jiang. Omnitokenizer: A joint image-video tokenizer for visual generation. *Advances in Neural Information Processing Systems*, 37:28281–28295, 2024. 2

[70] 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, 6

[71] Mark Weber, Lijun Yu, Qihang Yu, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. Maskbit: Embedding-free image generation via bit tokens. *arXiv preprint arXiv:2409.16211*, 2024. 4

[72] BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. 2, 3

[73] Junfeng Wu, Yi Jiang, Chuofan Ma, Yuliang Liu, Hengshuang Zhao, Zehuan Yuan, Song Bai, and Xiang Bai. Liquid: Language models are scalable and unified multi-modal generators. *arXiv preprint arXiv:2412.04332*, 2024. 2, 3

[74] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score

v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. 6

[75] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *arXiv preprint arXiv:2410.10629*, 2024. 6

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

[77] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024. 6

[78] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021. 2

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

[80] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023. 2, 3, 4

[81] Lijun Yu, José Lezama, Nitesh B 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. *arXiv preprint arXiv:2310.05737*, 2023. 2, 4, 5

[82] Lili Yu, Bowen Shi, Ramakanth Pasunuru, Benjamin Muller, Olga Golovneva, Tianlu Wang, Arun Babu, Binh Tang, Brian Karrer, Shelly Sheynin, et al. Scaling autoregressive multimodal models: Pretraining and instruction tuning. *arXiv preprint arXiv:2309.02591*, 2(3), 2023. 3

[83] Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *arXiv preprint arXiv:2406.07550*, 2024. 2

[84] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 3

[85] Yue Zhao, Yuanjun Xiong, and Philipp Krähenbühl. Image and video tokenization with binary spherical quantization. *arXiv preprint arXiv:2406.07548*, 2024. 2, 3, 4