Denoising Token Prediction in Masked Autoregressive Models

Ting Yao, Yehao Li, Yingwei Pan, Zhaofan Qiu, and Tao Mei
HiDream.ai Inc.
{tiyao, liyehao, pandy, qiuzhaofan, tmei}@hidream.ai

Abstract

Autoregressive models are just at a tipping point where they could really take off for visual generation. In this paper, we propose to model token prediction using diffusion procedure particularly in masked autoregressive models for image generation. We look into the problem from two critical perspectives: progressively refining the unmasked tokens prediction via a denoising head with the autoregressive model, and representing masked tokens probability distribution by capitalizing on the interdependency across masked and unmasked tokens through a diffusion head. Our proposal harbors an innate agency that remains advantageous in the speed of sequence prediction, and strongly favors high capability in generating quality samples by leveraging the principles of denoising diffusion process. Extensive experiments on both class-conditional and text-to-image tasks demonstrate its superiority, achieving the state-of-the-art FID score of 1.47 and 5.27 on ImageNet and MSCOCO datasets, respectively. More remarkably, our approach leads to 45% speedup in the inference time of image generation against the diffusion models such as DiT-XL/2.

1. Introduction

Autoregressive models are at the core of achieving unprecedented success of large language models (LLMs). The technologies follow the philosophy of Next Token Prediction that the models predict one token at a time in an autoregressive mechanism. Scaling up such autoregressive models effectively pushes the capabilities and limits in solving Natural Language Processing (NLP) tasks [3, 21, 36, 45, 48]. This tremendous progress encourages recent advances to delve into autoregressive models for visual generation tasks, e.g., text-to-image generation [5, 25, 26, 31, 42, 59] and text-to-video generation [55, 56]. Nevertheless, adopting autoregressive models in vision domain is not trivial and the difficulties generally originate from two aspects: 1) the prediction process of visual content in autoregressive form should be different from that of generating inherently sequential language; 2) vector quantization involved in typical autoregressive models to discretize images or videos may result in severe information loss. That somewhat affects the impact of scaling autoregressive models to vision data.

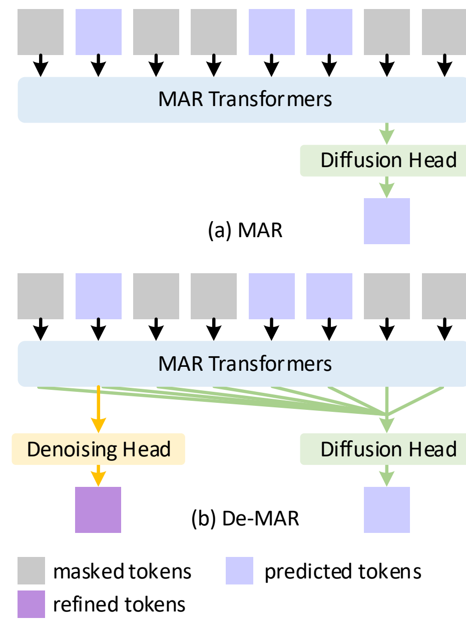

Figure 1. Illustration of (a) Masked Autoregressive (MAR) Model and (b) our token Denoising in MAR model (De-MAR), which contains a new denoising head for progressively refining unmasked tokens and an upgraded diffusion head with enhanced context information for masked token prediction.

In response to the above issues, we exploit the recipe of Masked Autoregressive (MAR) model in which all unmasked tokens are available to each other and also to all masked tokens. As such, the MAR model nicely leverages global interactions across tokens in the sequence and allows us to predict multiple tokens simultaneously in each step. In contrast to tokenizing sequentially in raster order which is widely-used in GPT-like structure for language prediction, the MAR manner might be more fit for modelling vision data. More importantly, we devise a noising-denoising process depicted as *denoising head* in Figure 1 (b) to further improve the predicted tokens to better match the data. To mitigate the second issue, we remould the autoregressive model on continuous-valued domains in [26] to represent the distribution for each masked token. Unlike taking only the token to be predicted as the condition in [26], all the masked and predicted tokens act as the condition of a diffusion procedure denoted as *diffusion head* in Figure 1 (b) to model masked tokens. By doing so, the model considers cross-attention capabilities across all the tokens, further enhancing the interdependency.

To consolidate the idea of modelling tokens using denoising mechanism, we present a new token Denoising approach in Masked Autoregressive models, namely De-MAR, for image generation. Specifically, De-MAR first exploits the pre-trained variational autoencoder (VAE) [22] to encode the input image into a sequence of visual tokens during training. The visual tokens are randomly selected and replaced with masked tokens. De-MAR then injects Gaussian noise to the unmasked tokens following the diffusion process in DDPM [18]. These tokens along with the context tokens (e.g., class tokens for class-conditional image synthesis or text tokens for text-to-image task) are fed into MAR Transformers to model the bidirectional relationship among tokens and output the conditional tokens for both unmasked token refinement and masked token prediction. For each unmasked noisy token, the denoising head takes the unmasked token and its conditional token as the input, and removes the added Gaussian noise. For the masked tokens, the Transformer-based diffusion head is devised to learn the reverse process of a fixed Markov Chain and model the visual token probability distribution via facilitating the contributions across all the conditional tokens. At inference, De-MAR employs the "next set-of-tokens prediction" scheme and progressively reduces the masking ratio as in MAR [26]. The diffusion head takes the tokens generated by MAR Transformer as a conditioning for a reverse diffusion process to predict the masked tokens, while the denoising head refines the predicted tokens at the previous step to better match the data. Once all the tokens are predicted, the VAE maps the token sequence into image pixels.

The main contribution of this work is the proposal of denoising token prediction in MAR framework for boosting up autoregressive models on visual generation. This issue also leads to the elegant views of how to permute and optimize token prediction with autoregressive process, and how to facilitate the contributions across all the tokens for modelling token probability distribution. We properly analyze and demonstrate the effectiveness of our De-MAR through extensive experiments on ImageNet and MSCOCO datasets to validate the great potential of autoregressive models in solving image generation tasks, showing superior performances and faster speed than Diffusion models.

2. Related Work

Image Generation with Diffusion Models. Diffusion models [18, 19, 33, 35, 39, 49, 62] frame image generation as an iterative denoising process that gradually creates images from random Gaussian noise. Existing diffusion models can be briefly grouped into two configurations: Convolutional U-Net (e.g., DALL-E 2 [37], Imagen [40], and Stable Diffusion [39]) and Diffusion Transformer (e.g., U-ViT [1], PixArt-α [7], SD3 [12], and HiDream-I1 [4]). In the first configuration, the latent-based approaches like Stable Diffusion commonly leverage VAE to compress and decode images within latent space, while pixel-based methods (e.g., Imagen) directly decode images in the pixel space. SDXL [34] further upgrades Stable Diffusion with three times larger U-Net structure and additional image-to-image refinement module. Replacing Convolutional U-Net structure with Transformer blocks [32, 51, 52] in the second configuration, Diffusion Transformer integrates time, noisy image patches, and specific condition as tokens under diffusion probabilistic scheme, showing remarkable scaling property. For example, PixArt-α presents an efficient large-scale Diffusion Transformer structure via decomposed three-step training strategy. SD3 further scales up Diffusion Transformer in rectified flow formulation with an improved noise samplers. REPA [57] designs a straight-forward regularization to ease the training of Diffusion Transformer by aligning the hidden state of noisy inputs with self-supervised visual representation. Diffusion forcing [6] proposes a new training paradigm that allows each token to be associated with different level of noise. Nevertheless, the computational cost of image generation with diffusion model is usually expensive owing to the inherent time-consuming iterative nature.

Image Generation with Autoregressive Models. Inspired by unprecedented success of autoregressive models in Large Language Models (LLMs) [3, 21, 36, 45, 48], some pioneering works start to exploit similar autoregressive paradigm for image generation, pursuing the potential of unified models for general artificial intelligence. In analogy to next token prediction in LLMs, the earlier attempts (e.g., VQ-VAE [38, 47], VQGAN [11, 24], Parti [54]) convert images into discrete tokens, and then decode image tokens via autoregressive models. Subsequently, MAR [26] novelly enables masked autoregressive modeling through a diffusion loss within a continuous-valued space, instead of conventional discrete-valued space. DART [14] leverages the full generative trajectory to iteratively denoise image patches in autoregressive manner. VAR [44] proposes another kind of autoregressive learning, i.e., next-scale prediction, that autoregressively produces multi-scale image token maps in a coarse-to-fine manner. LlamaGen [42] capitalizes on Llama architecture [45] as autoregressive model for image generation, aiming to reduce inductive biases on visual inputs via such potential unified vision-language model. The concurrent Open-MAGVIT2 [30] also explores Llama architecture for autoregressive image generation, and facilitates token prediction on super-large vocabulary with "next sub-token prediction" scheme.

However, most autoregressive models restrict sequential learning in a fixed raster order, resulting in sub-optimal uni-directional image token prediction for image generation. In this work, we start from masked autoregressive model that enables bidirectional interactions among tokens in the sequence, which might be more tailored for modeling image data. We go one step further by designing a novel denoising head for progressively refining predicted tokens and a diffusion head that strengthens masked tokens probability distribution with enhanced context from all tokens.

3. Approach: De-MAR

In this work, we devise a token Denoising in MAR model (De-MAR) that frames token prediction with diffusion procedure in the realm of masked autoregressive models. This section starts with an introduction of the overview architecture of De-MAR at training and inference stages. Then, a brief review of MAR Transformer structure widely adopted in masked autoregressive models is provided. After that, two unique designs for unmask and mask token predictions, i.e., denoising head and diffusion head, are elaborated. Figure 2 (a) shows an overview of De-MAR.

3.1. Overview

Suppose we have an image $I \in \mathbb{R}^{H\times W\times 3}$, De-MAR first exploits the pre-trained variational autoencoder (VAE) provided by MAR [26] to encode I into latent representations $I' \in \mathbb{R}^{h\times w\times d}$, where $h/w/d$ denotes the dimension of height/width/channels for the latent. Then, the latent representations are reshaped into a sequence of $N = h \cdot w$ visual tokens X, which are fed into MAR Transformer with denoising and diffusion heads for training.

At inference, De-MAR employs the "next set-of-tokens prediction" scheme through diffusion head by progressively reducing the masking ratio, and meanwhile leverages denoising head to gradually refine the predicted tokens. Finally, after all the tokens being predicted, the token sequence are mapped into image pixels by VAE decoder for image generation.

3.2. MAR Transformer

Following the typical architecture in Masked autoregressive models [5, 26, 31], MAR Transformer is composed of $\tilde{M}$ Transformer block. As shown in Figure 2 (b), each block consists of a self-attention layer and a feed-forward layer. During training, a masking ratio r is randomly sampled from a pre-defined distribution $p(r)$ within the $[0,1]$ range. Then, we randomly select $\lceil r\cdot N\rceil$ tokens and replace them with masked token $[MASK]$ in the sequence x. For the unmasked tokens x_u, De-MAR injects noise into them as in DDPM: $\tilde{x}_u^t = \sqrt{\bar{a}_t}x_u + \sqrt{1-\bar{a}_t}\varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0,\mathbf{I})$ and constant $\bar{a}_t$ defines the noise schedule hyperparameters. Next, the resultant sequence along with the context tokens X' (e.g., class tokens or text tokens) are fed into MAR Transformer. MAR Transformer models the bidirectional relationship among tokens, and produces the conditional tokens for both unmasked token refinement and masked token prediction. Specifically, the i-th Transformer block in MAR Transformer operates as:

$$\begin{aligned} \tilde{z}^i &= z^i + \mathbf{SelfAttn}(\mathbf{norm}(z^i)), \\ z^{i+1} &= \tilde{z}^i + \mathbf{FFN}(\mathbf{norm}(\tilde{z}^i)), \end{aligned} \tag{1}$$

where z^i is the input of the i-th Transformer block, $\mathbf{norm}$, $\mathbf{SelfAttn}$ and $\mathbf{FFN}$ denotes the layer normalization, self-attention layer, and feed-forward layer, respectively. Note that the input of the first Transformer block is $z^0 = WX'$, where W is the parameter that transforms the input sequence X' into the same dimension as MAR Transformer. The outputs of the final Transformer block $z = z^{\tilde{M}}$ are regarded as conditional tokens and will be fed into the denoising head and diffusion head for token prediction.

3.3. Denoising Head

The typical MAR adopts MAR Transformer to refine masked tokens with context information derived from all predicted/unmasked tokens, which acts as the condition of an independent diffusion head. Nevertheless, during training, the ground-truth tokens are commonly taken as input unmasked tokens of MAR Transformer to trigger autoregressive modeling. This inevitably results in a discrepancy of inputs at inference, since no ground-truth token is available and we have to take the predicted noisy tokens as the input unmasked tokens of MAR Transformer. To mitigate such training-inference discrepancy, we inject noises into ground-truth tokens and take such noisy tokens as the inputs of MAR Transformer during training, aiming to better mimic the inputs at inference. Meanwhile, a novel denoising head is devised to further refine the unmasked noisy tokens progressively, thereby encouraging better alignment of ground-truth tokens.

Technically, we implement denoising head as two parallel branches: token refinement branch and token assessment branch (see Figure 2 (c)). In particular, the token refinement branch is used to refine the unmasked noisy tokens. At the same time, the token assessment branch aims to adaptively modulate the refinement by dynamically fusing the predicted token and the refined one. Here the token refinement branch is implemented as a two-layer MLP that takes the unmasked noisy token x_i' and its conditional token z_i as inputs, yielding the refined token y_i as follows:

$$\begin{aligned} o_i^r &= \sigma([x'_i, z_i]W_1^r + b_1^r), \\ y_i &= o_i^r W_2^r + b_2^r + x'_i, \end{aligned} \tag{2}$$

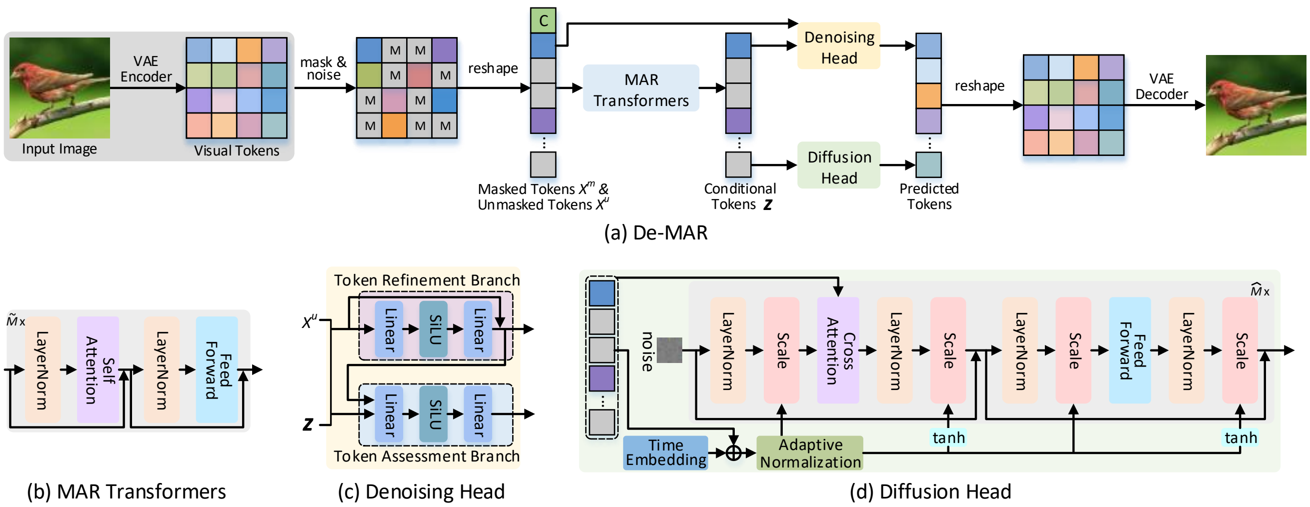

Figure 2. (a) An overview of De-MAR model. Given an input image, VAE encoder is first utilized to transform inputs into a sequence of visual tokens, some of which are further randomly replaced with masked tokens. Note that unmasked tokens are augmented with Gaussian noise to trigger noising-denoising process along with autoregressive modeling. After that, both unmasked and masked tokens (plus context tokens) are fed into MAR Transformer for mining bidirectional relations among all tokens. Next, for unmasked noisy tokens, a denoising head is devised to predict primary unmasked tokens without noise. Meanwhile, a Transformer-based diffusion head is utilized to capture richer context information among all tokens to facilitate masked token prediction. All predicted tokens are further reshaped and transformed into images via VAE decoder. (b-d) shows the detailed architectures of MAR Transformer, Denoising head, and Diffusion head.

where W_1^r, W_2^r, b_1^r, b_2^r are the parameters, $[\cdot]$ denotes the concatenation operation along channel dimension, and σ refers to the $SiLU$ activation [10]. During training, the token refinement branch is optimized by minimizing the difference between the refined token and its corresponding ground-truth token x_i without noise:

$$loss_r = \frac{1}{N^u} \sum\nolimits_{i=0}^{N^u} \|y_i - x_i\|^2, \tag{3}$$

where N^u is the number of unmasked tokens.

Similar to the token refinement branch, the token assessment branch takes the unmasked noisy token x_i', the refined token y_i and its conditional token z_i as inputs, and utilizes a two-layer MLP to estimate the quality score of x_i' and y_i:

$$\begin{aligned} o_i^a &= \sigma([x'_i, y_i, z_i]W_1^a + b_1^a), \\ s_i &= o_i^a W_2^a + b_2^a, \\ s_i^{x'}, s_i^y &= \mathbf{split}(s_i), \end{aligned} \tag{4}$$

where W_1^a, W_2^a, b_1^a, b_2^a are the parameters, **split** denotes the split operation along channel dimension, and $s_i^{x'}/s_i^y$ represents the estimated quality score of x_i'/y_i. During training, the token assessment branch is optimized with supervision of ground-truth quality scores for x_i' and y_i (i.e., the similarity between x_i'/y_i and the corresponding ground-truth token x_i). Specifically, we first measure the cosine similarity between x_i'/y_i and ground-truth token x_i by:

$$\begin{aligned} \tilde{s}_i^{x'} &= \mathbf{Similarity}(x_i', x_i), \\ \tilde{s}_i^y &= \mathbf{Similarity}(y_i, x_i), \end{aligned} \tag{5}$$

where **Similarity** denotes the cosine similarity. After that, the token assessment branch is thus optimized by minimizing the discrepancy between the estimated quality scores and ground-truth ones:

$$loss_a = \frac{1}{N^u} \sum\nolimits_{i=0}^{N^u} abs(s_i^{x'} - \tilde{s}_i^{x'}) + abs(s_i^y - \tilde{s}_i^y). \tag{6}$$

At inference, the token assessment branch estimates the quality scores for the input unmasked token and the refined unmasked token. Then, the aggregation weights are determined via normalization of the corresponding quality scores relative to their summation. The output unmasked token is finally obtained by linearly fusing the input and refined tokens with these aggregation weights.

3.4. Diffusion Head

Recall that MAR leverages a MLP-based diffusion head to learn the reverse process of a fixed Markov Chain and model the masked token probability distribution solely conditioned on the specific output masked token of MAR Transformer. Such way leaves the interdependency among all unmasked and masked tokens under-exploited, resulting in a sub-optimal solution for masked token prediction. To alleviate this limitation, we design a new Transformer-based diffusion head by exploiting the cross-attention capabilities across all unmasked and masked conditional tokens to facilitate masked token prediction.

Suppose we have the ground-truth token $x_i \in \mathbb{R}^d$ at the position i that will be predicted. The goal of diffusion head is to model the probability distribution of x_i. Unlike the conventional MAR [26] that only capitalizes on its corresponding conditional token z_i as inputs, our Transformer-based diffusion head takes the whole unmasked and masked tokens $z = [z_i]_{i=0}^{N-1}$ as conditions, which reflect the global image context across all tokens. Here the objective of diffusion head can be defined as a standard denoising process:

$$\mathcal{L}(z, x_i) = \mathbb{E}_{\varepsilon,t}\left[\left\|\varepsilon - \varepsilon_\theta(x_i^t|t, z)\right\|^2\right], \tag{7}$$

where $\varepsilon \in \mathbb{R}^d$ is the noise vector sampled from distribution $\mathcal{N}(0, \mathbf{I})$, x_i^t is the noise-corrupted vector formulated as $x_i^t = \sqrt{\bar{a}_t}x_i + \sqrt{1-\bar{a}_t}\varepsilon$, t is a time step of the noise schedule, ε_θ is the noise estimator (θ: the parameters of diffusion head). In this way, the diffusion head takes x_i^t as the input and estimates the noise conditioned on both t and z.

Technically, we implement this Transformer-based diffusion head as a stacked of $\hat{M}$ Transformer blocks (see Fig 2 (d)). Each Transformer block consists of a cross-attention layer, a feed-forward layer and an adaptive normalization layer. Nevertheless, the linear projections in the adaptive normalization layer occupy a substantial proportion of parameters as pointed by [7]. To significantly reduce the parameters of diffusion head, we share the parameters of the adaptive normalization layer across all blocks. Specifically, for the i-th token, the shared adaptive normalization layer firstly takes its conditional masked token z_i and the time embedding te as inputs, and generates the context vector c_i as follows:

$$\begin{aligned}\hat{c}_i &= z_i + te,\\ c_i &= W_c\sigma(\hat{c}_i) + b_c,\end{aligned} \tag{8}$$

where W_c and b_c are the parameters, σ denotes the $SiLU$ activation. After that, conditioned on all conditional tokens z and the context vector c_i, the j-th Transformer block for the input masked token $x_{i,j}$ operates as:

$$\begin{aligned}
z^n &= \mathbf{norm}(z),\\
\tilde{c}_i &= \mathbf{scale}\cdot c_i + \mathbf{shift},\\
\alpha_1, \beta_1, \alpha_2, \beta_2 &= \mathbf{split}(\tilde{c}_i),\\
y_i^a &= \mathbf{CrossAttn}(\alpha_1\cdot\mathbf{norm}(x_{i,j}), z^n, z^n),\\
y_i &= x_{i,j} + \tanh(\beta_1)\cdot\mathbf{norm}(y_i^a),\\
y_i^f &= \mathbf{FFN}(\alpha_2\cdot\mathbf{norm}(y_i)),\\
x_{i,j+1} &= y_i + \tanh(\beta_2)\cdot\mathbf{norm}(y_i^f),
\end{aligned} \tag{9}$$

where $\mathbf{scale}$ and $\mathbf{shift}$ are the parameters, $\mathbf{split}$ denotes the split operation along channel dimension, and $\mathbf{CrossAttn}$ refers to the cross-attention layer. Note that the input of the first Transformer block is the noise-corrupted vector $x_{i,0} = x_i^t$.

At inference, the diffusion head starts from $x_i^T \sim \mathcal{N}(0, \mathbf{I})$, and conducts the reverse diffusion procedure by:

$$x_i^{t-1} = \frac{1}{\sqrt{a_t}}(x_i^t - \frac{1-a_t}{\sqrt{1-\bar{a}_t}}\varepsilon_\theta(x_i^t|t, z)) + \sigma_t\delta, \tag{10}$$

Table 1. The architecture configurations of the family of De-MAR in three different scales (i.e., Base, Large, and Huge).

Model	#Layers	Hidden size	MLP size	#Heads	#Params
De-MAR-B	24	768	3,072	12	227M
De-MAR-L	32	1,024	4,096	16	524M
De-MAR-H	40	1,280	5,120	16	995M

where σ_t denotes the noise level at the time step t, and δ is the noise sampled from Gaussian distribution $\mathcal{N}(0, \mathbf{I})$. Finally, the diffusion head predicts all masked tokens for image generation.

4. Experiments

4.1. Datasets

We empirically verify the merit of De-MAR for image generation in comparison with state-of-the-art approaches on two datasets of ImageNet [8] and MS-COCO [27]. The first experiment validates De-MAR on ImageNet at 256×256 resolution for class-conditional image generation. This dataset contains 1,281,167 training images from 1K different classes. The second experiment evaluates De-MAR on MS-COCO at 256×256 resolution for text-to-image generation. MS-COCO is composed of 82,783 training images and 40,504 validation images. Each image is annotated with five captions. To enable image generation from texts, we follow Stable Diffusion [39] and convert discrete captions into a sequence of text embeddings by using CLIP text encoder. After that, the resultant text embeddings are regarded as the sequence of tokens, which will be fed into De-MAR for autoregressive modeling.

4.2. Experimental Settings

Here we provide the details of network architectures, training setup, and evaluation metrics. More detailed configurations are included in supplementary material.

Network Architectures. Following the architecture configurations of MAR family [26], we build three variants of our De-MAR in three different scales (i.e., Base, Large, and Huge). Similar to MAR-B/L/H, the number of Transformer blocks in MAR Transformer of De-MAR-Base (De-MAR-B), De-MAR-Large (De-MAR-L), and De-MAR-Huge (De-MAR-H) is 24, 32, and 40, respectively. The number of Transformer blocks in diffusion head of De-MAR-B/L/H is set as 4/6/9. Table 1 details the configurations (e.g., layer number, hidden size, MLP size) of three De-MAR variants.

Training Setup. At training stage, we conduct all experiments on 32× 80GB-A100 GPUs. For class-conditional image generation on ImageNet, we follow MAR [26] and randomly sample masking ratio in [0.7, 1.0]. We adopt the common KL-16 version of tokenizer in MAR [26]. At inference, the masking ratio is progressively reduced from 1.0 to 0 via cosine schedule. We use the AdamW [28] op-

Table 2. Performance comparison with state-of-the-art approaches on ImageNet for class-conditional image generation. We group all approaches into four categories: Generative Adversarial Network based methods (GAN), Diffusion methods (Diffusion), Autoregressive Models (AR), and Masked autoregressive models (Mask.). CFG denotes the use of classifier-free guidance [17]. Note that most methods are constructed in various model sizes, and thus we report the specific parameter number (#Para.) of each run. Three variants of our De-MAR in base, large, and huge scales are included.

			w/o CFG				w/ CFG			
Type	Model	#Para.	FID ↓	IS ↑	Precision ↑	Recall ↑	FID ↓	IS ↑	Precision ↑	Recall ↑
GAN	BigGAN [2]	112M	6.95	224.5	0.89	0.38	-	-	-	-
	GigaGAN [20]	569M	3.45	225.5	0.84	0.61	-	-	-	-
	StyleGan-XL [41]	166M	2.30	265.1	0.78	0.53	-	-	-	-
Diffusion	ADM [9]	554M	10.94	101.0	0.69	0.63	4.59	186.7	0.82	0.52
	CDM [19]	-	-	-	-	-	4.88	158.7	-	-
	LDM-4 [39]	400M	10.56	103.5	0.71	0.62	3.60	247.7	0.87	0.48
	U-ViT-H/2-G [1]	501M	-	-	-	-	2.29	263.9	0.82	0.57
	DiT-XL/2 [33]	675M	9.62	121.5	0.67	0.67	2.27	278.2	0.83	0.57
AR	VQGAN [11]	227M	18.65	80.4	0.78	0.26	-	-	-	-
	VQGAN-re [11]	1.4B	5.20	280.3	-	-	-	-	-	-
	ViT-VQGAN-re [53]	1.7B	3.04	227.4	-	-	-	-	-	-
	GIVT [46]	304M	5.67	-	0.75	0.59	3.35	-	0.84	0.53
	LlamaGen-L [42]	343M	19.07	64.3	0.61	0.67	3.07	256.1	0.83	0.52
	LlamaGen-XL [42]	775M	15.54	79.2	0.62	0.69	2.62	244.1	0.80	0.57
	LlamaGen-XXL [42]	1.4B	14.65	86.3	0.63	0.68	2.34	253.9	0.80	0.59
Mask.	MaskGIT [5]	227M	6.18	182.1	0.80	0.51	-	-	-	-
	Token-Critic [25]	422M	4.69	174.5	-	-	-	-	-	-
	AutoNAT-L [31]	194M	-	-	-	-	2.68	278.8	-	-
	MAR-B [26]	208M	3.48	192.4	0.78	0.58	2.31	281.7	0.82	0.57
	MAR-L [26]	479M	2.60	221.4	0.79	0.60	1.78	296.0	0.81	0.60
	MAR-H [26]	943M	2.35	227.8	0.79	0.62	1.55	303.7	0.81	0.62
	De-MAR-B	227M	3.14	200.3	0.78	0.59	2.04	283.5	0.82	0.59
	De-MAR-L	524M	2.48	226.9	0.80	0.60	1.66	300.5	0.82	0.60
	De-MAR-H	995M	2.19	228.1	0.80	0.63	1.47	305.8	0.83	0.62

timizer for training, where the learning rate is set as 8e-4 and the weight decay is 0.02. We set the batch size as 2,048. For text-to-image generation on MS-COCO, we follow AutoNAT-L [31] and randomly sample masking ratio according to Beta distribution where $\alpha = 3$ and $\beta = 1$. The AdamW optimizer is adopted for training with weight decay of 0.03 (learning rate: 2e-4, batch size: 256).

Evaluation Metrics. For evaluation, we leverage Fréchet inception distance (FID) [16] as the main metric on both ImageNet and MS-COCO. Moreover, we use Inception Score (IS) [29] and Precision/Recall [23] as secondary metrics on ImageNet. For fair comparison, we strictly follow MAR [26] and implement all evaluation metrics via ADM's TensorFlow scripts [9].

4.3. Results on Class-conditional Image Generation

Table 2 summarizes the performance comparisons of different generative models for class-conditional image generation on ImageNet dataset. Note that all runs can be briefly grouped into four categories: the conventional Generative Adversarial Network (GAN) based methods, diffusion-based methods, autoregressive models, and masked autoregressive models. Except for the typical GAN based methods, we report the performances of most runs in the rest three categories under two different inference settings, i.e., with or without Classifier-Free Guidance (CFG).

Overall, under each setting (with or without CFG), the results in terms of most metrics consistently indicate that our proposed De-MAR achieves superior performances against existing state-of-the-art approaches derived from all four categories with comparable sizes. In particular, the FID score of our De-MAR-B in base scale reaches 2.04 with CFG, making the absolute improvement over the best competitor in AR (i.e., MAR-B) by 0.27. The results generally highlight the key advantage of exploiting diffusion procedure to model token prediction in masked autoregressive models. As expected, the typical GAN-based method (BigGAN) fails to achieve high image fidelity on large unstructured dataset, and StyleGan-XL further scales up StyleGAN3 with a progressive growing strategy, leading to impressive performances. For diffusion model category, different from latent diffusion model with conventional U-Net structure, U-ViT and DiT adopt Transformer structure to build diffusion model, showing remarkable scaling property with superior performances than all GAN based methods.

Table 3. Performance comparison with state-of-the-art approaches on MS-COCO for text-to-image generation. We group all approaches in three categories: GAN based methods (GAN), Diffusion methods (Diffusion), and Masked AR models (Mask.).

Type	Model	FID ↓
GAN	AttnGAN [50]	35.49
	DM-GAN [61]	32.64
	DF-GAN [43]	19.32
	XMC-GAN [58]	9.33
	LAFITE [60]	8.12
Diffusion	VQ-Diffusion [15]	19.75
	Friro [13]	8.97
	U-ViT-S/2 (Deep) [1]	5.48
Mask.	AutoNAT-S [31]	5.36
	MAR [26]	6.36
	De-MAR	5.27

Note that the use of CFG significantly boosts up the FID, IS, and Precision scores of most diffusion and AR models across varied scales. But the corresponding Recall scores commonly decrease, and we speculate that CFG might sacrifice diversity. In comparison to diffusion models, autoregressive models (e.g., VQGAN and LlamaGen) seek a different way of generative modeling by regrading images as a sequence of discrete tokens, and obtain comparable image generation results. Furthermore, the mask autoregressive models introduce mask tokens into autoregressive modeling. This way generally facilitates contextual information/representation learning along generative modeling and thus improves performances. Specifically, instead of typically accompanying autoregressive modeling with discrete tokenizer (e.g., LlamaGen and AutoNAT-L), MAR adopts continuous tokenizer via a diffusion loss in the realm of masked autoregressive modeling, leading to large performance boosts. But the performances of MAR across different model sizes are still inferior to our De-MAR. This confirms the effectiveness of the denoising head for progressively refinement of unmasked tokens and the diffusion head for enhanced masked token prediction with richer context information among all tokens. When taking an in-depth look at three variants of our De-MAR in various model sizes, an encouraging scaling behavior is attained. Such observation aligns well with autoregressive language models (e.g., GPT) and autoregressive vision models (e.g., LlamaGen and MAR), which again validates the promising scalability of autoregressive models.

4.4. Results on Text-to-Image Generation

Next, we show the performance comparisons on MS-COCO for text-to-image generation in Table 3. Note that for fair comparison, we build a light-weight version of our De-MAR (including 16 Transformer blocks in MAR Transformer) with comparable model size as most baselines. Here we group all runs in three categories, i.e., GAN based models, diffusion models, and masked autoregressive models. In general, our De-MAR performs consistently better than all baselines on this challenging dataset. In particular, the FID score of De-MAR can reach 5.27, which makes the absolute improvement over the best competitor AutoNAT-S by 0.09. Compared to most GAN based approaches, diffusion models with convolutional network structure (e.g., VQ-Diffusion and Friro) exhibit more flexible and scalable generative modeling, and obtain superior performances for text-to-image generation. By replacing convolutional network structure with Transformer structure, U-ViT achieves lower FID scores, basically validating the benefit of Diffusion Transformer as a higher-capacity backbone. Similar to the observations on ImageNet, there is a performance gap between MAR and our De-MAR. Though both runs belong to masked autoregressive models with continuous tokenizer, De-MAR upgrades MAR with an upgraded diffusion head that additionally mines context among all tokens and a new denoising head to progressively refine unmasked tokens, yielding apparent improvements.

Table 4. Performance comparison on ImageNet for class-conditional image generation across different ablated runs of our De-MAR-B, i.e., the base model of typical MAR with MLP-based diffusion head, MAR with an upgraded Transformer-based Diffusion Head (**Diff. Head**), MAR with additional Denoising Head (**De. Head**), and the full version of De-MAR.

Diff. Head	De. Head	FID ↓	IS ↑	Precision ↑	Recall ↑
		2.31	281.7	0.82	0.57
✓		2.10	283.1	0.82	0.58
	✓	2.22	282.4	0.82	0.57
✓	✓	2.04	283.5	0.82	0.59

4.5. Experimental Analysis

Ablation Study. Here we study how each design in our De-MAR influences the overall performances. Recall that our De-MAR remoulds the typical MAR with two novel designs, i.e., denoising head (**De. Head**) that progressively refines unmasked tokens and Transformer-based diffusion head (**Diff. Head**) that strengthens masked token prediction with richer context. Table 4 lists the performances across different ablated runs of our De-MAR-B on ImageNet for class-conditional image generation. In particular, we start from the base model of typical MAR (the first row in this table), which equips MAR Transformer with a MLP-based diffusion head. Next, by additionally mining the richer context among all tokens for masked token prediction, MAR with **Diff. Head** (the second row) exhibits better performances. Meanwhile, we also include an ablated run by upgrading MAR solely with denoising head. That is, MAR with **De. Head** (the third row) injects noise into the input unmasked tokens of MAR Transformer, and adds a denoising head to further refine predicted tokens progressively. Compared to MAR, this way nicely alleviates the training-inference discrepancy and encourages predicted tokens better aligned with ground-truth data, thereby leading to bet-

Figure 3. Qualitative comparisons between MAR and our De-MAR on (a) ImageNet for class-conditional image generation and (b) MS-COCO for text-to-image generation. We adopt De-MAR-H on ImageNet, and a light-weight version of De-MAR on MS-COCO.

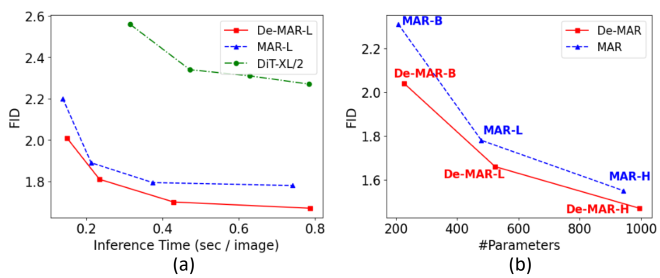

Figure 4. Computational cost ((a) inference time and (b) parameter number) vs. FID score on ImageNet.

ter performances. The full version of our De-MAR (the fourth row) is finally benefited from both **Diff. Head** and **De. Head**, and achieves the best performances across most metrics. The results basically validate the complementarity of our designed two heads.

Qualitative Results. To qualitatively evaluate our De-MAR, we showcase eight image results generated by MAR and De-MAR on ImageNet and MS-COCO in Figure 3. We clearly observe that De-MAR generates higher-quality images with less distortions and better aligned semantics with input class/caption, verifying the effectiveness of denoising token prediction in masked autoregressive models.

Computational Cost Analysis. Here we evaluate our De-MAR family with regard to computational cost (parameter number and inference time) and FID score on ImageNet. Figure 4 (a) shows the inference time-FID curve by varying the number of autoregressive/diffusion steps for each method, and Figure 4 (b) depicts the parameter number-FID curve for the family of each method across different scales. As shown in the figure, the curves of our De-MAR are always below the ones of other approaches. In other words, our De-MAR family seeks better computational cost-FID tradeoffs than existing methods.

5. Conclusions

In this work, we present the token Denoising approach in Masked Autoregressive models (namely De-MAR), a new generative model to facilitate both unmasked and masked token prediction in autoregressive modeling. De-MAR capitalizes on a new denoising head to progressively refine the unmasked tokens prediction, pursuing better alignment with true data. Additionally, a Transformer-based diffusion head is devised to mine cross-attention capabilities across all tokens, thereby strengthening masked token prediction with richer context information. Extensive experiments validate that De-MAR manages to boost up performances on both ImageNet and MS-COCO datasets for image generation.

Acknowledgement. This work was supported in part by the Beijing Municipal Science and Technology Project No. Z241100001324002 and Beijing Nova Program No. 20240484681.

References

[1] 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 *CVPR*, 2023. 2, 6, 7

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. arxiv 2018. *arXiv preprint arXiv:1809.11096*, 1809. 6

[3] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 1, 2

[4] Qi Cai, Jingwen Chen, Yang Chen, Yehao Li, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Yiheng Zhang, Fengbin Gao, Peihan Xu, et al. Hidream-i1: A high-efficient image generative foundation model with sparse diffusion transformer. *arXiv preprint arXiv:2505.22705*, 2025. 2

[5] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *CVPR*, 2022. 1, 3, 6

[6] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *NeurIPS*, 2024. 2

[7] 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*, 2023. 2, 5

[8] 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. 5

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 6

[10] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 2018. 4

[11] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 2, 6

[12] 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. 2

[13] Wan-Cyuan Fan, Yen-Chun Chen, DongDong Chen, Yu Cheng, Lu Yuan, and Yu-Chiang Frank Wang. Frido: Feature pyramid diffusion for complex scene image synthesis. In *AAAI*, 2023. 7

[14] Jiatao Gu, Yuyang Wang, Yizhe Zhang, Qihang Zhang, Dinghuai Zhang, Navdeep Jaitly, Josh Susskind, and Shuangfei Zhai. Dart: Denoising autoregressive transformer for scalable text-to-image generation. In *ICLR*, 2025. 2

[15] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*, 2022. 7

[16] 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. 6

[17] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 6

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 2022. 2, 6

[20] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *CVPR*, 2023. 6

[21] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019. 1, 2

[22] Diederik P Kingma. Auto-encoding variational bayes. In *ICLR*, 2013. 2

[23] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019. 6

[24] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *CVPR*, 2022. 2

[25] José Lezama, Huiwen Chang, Lu Jiang, and Irfan Essa. Improved masked image generation with token-critic. In *ECCV*, 2022. 1, 6

[26] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *arXiv preprint arXiv:2406.11838*, 2024. 1, 2, 3, 5, 6, 7

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 5

[29] Cheng Lu, Kaiwen Zheng, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Maximum likelihood training for score-based diffusion odes by high order denoising score matching. In *ICML*, 2022. 6

[30] 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. 2

[31] Zanlin Ni, Yulin Wang, Renping Zhou, Jiayi Guo, Jinyi Hu, Zhiyuan Liu, Shiji Song, Yuan Yao, and Gao Huang. Revisiting non-autoregressive transformers for efficient image synthesis. In *CVPR*, 2024. 1, 3, 6, 7

[32] Yingwei Pan, Yehao Li, Ting Yao, Chong-Wah Ngo, and Tao Mei. Stream-vit: Learning streamlined convolutions in vision transformer. *IEEE Transactions on Multimedia*, 2025. 2

[33] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *CVPR*, 2023. 2, 6

[34] 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. In *ICLR*, 2023. 2

[35] Yurui Qian, Qi Cai, Yingwei Pan, Yehao Li, Ting Yao, Qibin Sun, and Tao Mei. Boosting diffusion models with moving average sampling in frequency domain. In *CVPR*, 2024. 2

[36] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 1, 2

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

[38] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *NeurIPS*, 2019. 2

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 5, 6

[40] 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. In *NeurIPS*, 2022. 2

[41] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH*, 2022. 6

[42] 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. 1, 2, 6

[43] Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *CVPR*, 2022. 7

[44] 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

[45] 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. 1, 2

[46] Michael Tschannen, Cian Eastwood, and Fabian Mentzer. Givt: Generative infinite-vocabulary transformers. *arXiv preprint arXiv:2312.02116*, 2023. 6

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

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1, 2

[49] Siqi Wan, Jingwen Chen, Yingwei Pan, Ting Yao, and Tao Mei. Incorporating visual correspondence into diffusion model for virtual try-on. In *ICLR*, 2025. 2

[50] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 7

[51] Ting Yao, Yehao Li, Yingwei Pan, Yu Wang, Xiao-Ping Zhang, and Tao Mei. Dual vision transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2

[52] Ting Yao, Yehao Li, Yingwei Pan, and Tao Mei. Hiri-vit: Scaling vision transformer with high resolution inputs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[53] 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. In *ICLR*, 2022. 6

[54] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 2

[55] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *CVPR*, 2023. 1

[56] 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. 1

[57] 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. *arXiv preprint arXiv:2410.06940*, 2024. 2

[58] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Cross-modal contrastive learning for text-to-image generation. In *CVPR*, 2021. 7

[59] Guangting Zheng, Yehao Li, Yingwei Pan, Jiajun Deng, Ting Yao, Yanyong Zhang, and Tao Mei. Hierarchical masked autoregressive models with low-resolution token pivots. In *ICML*, 2022. 1

[60] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Towards language-free training for text-to-image generation. In *CVPR*, 2022. 7

[61] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *CVPR*, 2019. 7

[62] Rui Zhu, Yingwei Pan, Yehao Li, Ting Yao, Zhenglong Sun, Tao Mei, and Chang Wen Chen. Sd-dit: Unleashing the power of self-supervised discrimination in diffusion transformer. In *CVPR*, 2024. 2