Efficient-VQGAN: Towards High-Resolution Image Generation with Efficient Vision Transformers

Shiyue Cao[1,2*], Yueqin Yin[1,2*], Lianghua Huang[3], Yu Liu[3], Xin Zhao[1,2,4†], Deli Zhao[3], Kaiqi Huang[1,2,4]

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences
[2]Institute of Automation, Chinese Academy of Sciences, China
[3]Machine Intelligence Technology Lab,Alibaba Group, China
[4]CAS Center for Excellence in Brain Science and Intelligence Technology, China

caoshiyue2021@ia.ac.cn, {xuangen.hlh, ly103369}alibaba-inc.com,
xzhaopersonal@foxmail.com, {yinyueqin0314, zhaodeli}@gmail.com, kqhuang@nlpr.ia.ac.cn

Abstract

Vector-quantized image modeling has shown great potential in synthesizing high-quality images. However, generating high-resolution images remains a challenging task due to the quadratic computational overhead of the self-attention process. In this study, we seek to explore a more efficient two-stage framework for high-resolution image generation with improvements in the following three aspects. (1) Based on the observation that the first quantization stage has solid local property, we employ a local attention-based quantization model instead of the global attention mechanism used in previous methods, leading to better efficiency and reconstruction quality. (2) We emphasize the importance of multi-grained feature interaction during image generation and introduce an efficient attention mechanism that combines global attention (long-range semantic consistency within the whole image) and local attention (fined-grained details). This approach results in faster generation speed, higher generation fidelity, and improved resolution. (3) We propose a new generation pipeline incorporating autoencoding training and autoregressive generation strategy, demonstrating a better paradigm for image synthesis. Extensive experiments demonstrate the superiority of our approach in high-quality and high-resolution image reconstruction and generation.

1. Introduction

High-fidelity image synthesis has achieved promising performance thanks to the progress of generative models, such as generative adversarial networks (GANs) [12, 21, 22], diffusion models [15, 8] and autoregressive models [11, 44]. Moreover, high-resolution image generation, a vital generation task with many practical applications, provides better visual effects and user experience in the advertising and design industries. Some recent studies have attempted to achieve high-resolution image generation. Style-GAN [21, 22] leverages progressive growth to generate high-resolution images. However, GAN-based models often suffer from training stability and poor mode coverage [35, 48]. As diffusion models continue to evolve, recent studies [31, 34] have begun to explore the utilization of cascaded diffusion models for generating high-resolution images. This approach involves training multiple independent and enormous models to collectively accomplish a generation task. On another note, some researchers [11, 44, 5] leverage a two-stage vector-quantized (VQ) framework for image generation, which first quantizes images into discrete latent codes and then model the data distribution over the discrete space in the second stage. Nonetheless, under the limited computational resources (*e.g.*, memory and training time), the architectures of the existing vector-quantized methods are inferior. In this paper, to solve the problems of existing models, we would like to explore a more efficient two-stage vector quantized framework for high-resolution image generation and make improvements from the following three aspects.

Firstly, prior methods [11, 44] claim the importance of the attention mechanism in the first quantization stage for better image understanding, and they leverage global attention to capture long-range interactions between discrete tokens. However, we find this global attention not necessary for image quantization based on the observation that the alteration of several tokens will only influence their nearby tokens. Hence, local attention can yield satisfactory reconstruction results and circumvent the computationally in-

---

tensive nature of global attention, especially when generating high-resolution images. Consequently, we propose Efficient-VQGAN for image quantization adopting image feature extractor with local attention mechanism. This contributes to the acceleration of image reconstruction and dedicates more computation to the local information, further improving the reconstruction quality.

Besides, for the second stage of the existing vector-quantized methods [11, 44, 5], it would be intractable to generate high-resolution images since the quadratic space and time complexity is respected to the discrete sequence length. Further, the global self-attention interaction could lead to the insufficient ability to capture fine details in local areas. Accordingly, the fined-grained local attention at a token level for better local details capturing plays an essential role as coarse-grained global interaction for long-range context information capturing. We then utilize multi-grained attention, which implements different granularity of attention operations depending on the distance between tokens. As a result, it can support high-resolution image generation with a reduced length of the quantized image token sequence and reasonable computational cost.

Additionally, some recent studies related to text generation [43, 2] in the field of natural language processing, which combine the merits of autoencoding pretraining and autoregressive generation, show great potential in generating high-quality text sequence. Pretrained autoencoding models like BERT [23] can exploit bidirectional context to capture more information for reconstructing the masked input corpus, while autoregressive generation performing explicit density estimation can ensure consistency of output token sequence. Inspired by such combined training and inference strategy, we propose a similar pipeline for image generation tasks. In the training stage, we utilize an autoencoding-based masked visual token modeling strategy which is trained to recover the randomly masked image tokens by attending to tokens from all directions, better capturing contextual information. In the inference stage, combined with our block-based multi-grained attention mechanism, we autoregressively sample each image block in a fixed order and iteratively sample the tokens within the block in parallel, contributing to improved sampling speed and generation quality.

The contributions of this work can be summarized as follows. (1) We propose a more efficient two-stage vector-quantized framework with several improvements in the first quantization stage and the second generative modeling stage, yielding faster computational efficiency and better image quality. (2) We propose a new image generation pipeline that combines the advantages of autoencoding training and autoregressive generation, further improving the synthesis quality. (3) The proposed two-stage vector-quantized model demonstrates the capability to generate

higher-quality images at a faster speed on FFHQ and ImageNet datasets compared to previous methods.

2. Related Work

2.1. Image Synthesis

Recent development in generative modeling enables algorithms to generate high-quality and realistic images. Generative adversarial networks (GANs) facilitate image generation with promising results. However, GAN-based models [12, 21, 22] have poor mode coverage and struggle to model complex distributions due to the inductive priors imposed by convolutions. Diffusion models have received substantial attention these days, among which many attempts have been made at continuous diffusion models with remarkable results in image generation [8, 27] and image editing tasks [24, 1]. Some Transformer-based methods [6, 40, 29, 33] have shown strong power of density estimation using a fixed forward factorization order for image generation tasks. GAN-based methods[47, 18] incorporate Transformer with block-wise self-attention to scale up to higher-resolution images. In this paper, we are interested in exploiting two-stage vector quantized models based on vision transformers with multi-grained attention mechanism, which model the data distribution in compressed space and can increase the resolution of the generated images.

2.2. Vector-Quantized Image Modeling

Two-stage vector-quantized approaches will first utilize an image tokenizer to extract a discrete image token sequence. Then in the second stage, a generative model is trained to model the token distribution in the discrete latent space. VQ-VAE [40] tokenizes an image into discrete visual tokens by performing online clustering and then models token distribution autoregressively with a convolutional architecture. DALL·E [32] utilizes the first stage of VQ-VAE with Gumbel-Softmax strategy [17], and then uses an autoregressive Transformer-based likelihood model to generate visual tokens from the given text input. VQ-GAN [11] extends VQ-VAE by adding an adversarial and perceptual metric training loss in the first stage, producing higher-quality reconstructed images. Recently, ViT-VQGAN [44] improves VQGAN in architecture design and proposes to use a VIT backbone, yielding better reconstruction quality and inefficient. However, all those previous approaches employ an autoregressive model for image generation following the raster scan order, which is not efficient. MaskGIT [5] proposes a new image generation paradigm using a bidirectional Transformer decoder trained on Masked Visual Token Modeling(MVTM). In the inference stage, MaskGIT employs a non-autoregressive decoding strategy to produce images in a constant number of steps. However, these vector-quantized models still struggle
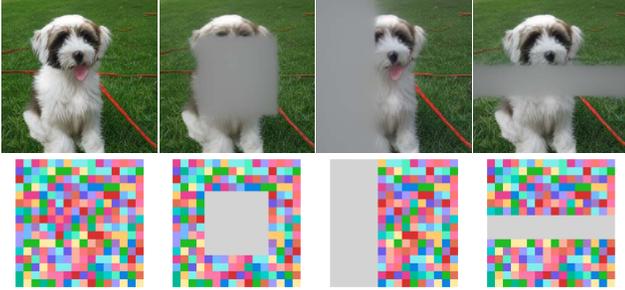
Figure 1. **The locality of image quantization.** When replacing a local region of latent codes encoded by ViT-VQGAN [44] with a specific token, only surrounding area is affected, while others remain the same.

to generate high-resolution images due to the design of sub-optimal two-stage architecture. Our model develops a more efficient framework for both stages, supporting the generation of high-resolution images.

3. Methodology

Our goal is to achieve high-resolution image generation through a more efficient two-stage vector-quantized image modeling framework, Efficient-VQGAN. Fig. 2 shows the structure of our model. For the first stage, instead of representing an image based on global attention, we design a more efficient vector-quantized model utilizing local attention-based encoder-decoder, as described in Sec. 3.1. For the second stage, we propose to learn a Transformer that incorporates global and local attention, which dramatically reduces the sequence length, tackling the quadratic dilemma in previous Transformer-based generative models, as described in Sec. 3.2. Additionally, we introduce a new training and inference paradigm for image synthesis, which performs masked visual token modeling in the training stage and autoregressive sampling in the generation stage, further improving the image quality, as described in Sec. 3.3.

3.1. Locality of Image Quantization

VQGAN [11] adds a non-local attention block in the encoder and decoder model, demonstrating the importance of the attention mechanism for better image understanding. ViT-VQGAN [44] replace the CNN encoder and decoder with Vision Transformer [9] using the global attention mechanism, which further improves the reconstruction quality. However, after conducting extensive experiments, we found that the global attention used in the previous image quantization process is not necessary because of the locality of image quantization (see Fig. 1), which instead increases the computational cost. Based on the observation, we propose our efficient vector-quantized autoencoder, performing local interaction based on Swin Transformer block [26] that has also shown outstanding performance in

other image recognition and dense prediction tasks [25, 4]. The overall architecture of the first stage consists of an encoder, a discrete codebook, a decoder, and a discriminator. Given an image $\boldsymbol{x} \in \mathbb{R}^{3 \times H \times W}$, the encoder will downsample it into a feature map $\boldsymbol{z} \in \mathbb{R}^{d \times \frac{H}{16} \times \frac{W}{16}}$. Then a discrete codebook is queried to produce a quantized image feature map z_q, and then we feed z_q to the decoder to reconstruct the original image.

Efficient-VQGAN Encoder. Given an input image, the encoder will first divide it into patches with fixed size (4×4), and then a linear embedding layer is applied to transform the patch feature into an arbitrary dimension. Then the patch feature will be passed through several Swin Transformer blocks to perform local attention, and achieve $2\times$ downsampling through patch merging layers. Downsampling factor f can be adjusted by adding or removing downsample blocks. Therefore, after being encoded by the Efficient-VQGAN encoder with default two downsample blocks, the resolution of the feature map is $1/16$ of the original image. Output feature map z produced by encoder will be passed through the quantized module introduced from VQGAN [11] to output quantized feature map z_q.

Efficient-VQGAN Decoder. The Efficient-VQGAN decoder and encoder are symmetrical. The decoder module also consists of several stages, totally implement upsample scale equal to f. In these stages, each Patch Expanding block achieve $2\times$ upsampling, and finally, a nearest neighbor interpolation upsample module is used to perform $2\times$ up-sampling to generate the reconstructed image.

Training loss of Image Quantization. To enable training process stable and convergent for high-resolution images, Projected GANs discriminator [36] is applied to produce GAN loss L_{Adv}, discriminating samples in deep feature space by using pretrained image feature extraction model, leading to improvement in quality and convergence speed. Besides, perceptual loss [19] is applied to enhance details and perceptual quality. The pixel level l_2 loss between input and reconstructed images and vector quantization loss [11] L_{VQ} are also introduced. The total loss defined as follow:

$$L = L_{Perceptual} + L_2 + L_{VQ} + \lambda * L_{Adv} \qquad (1)$$

where λ is adaptive weight [11].

3.2. Multi-Grained Attention for Efficient Image Generation

The quadratic computational complexity of the number of visual tokens limits the capability of self-attention Transformer based-models to generate high-resolution images. As mentioned in Sec. 3.1, applying global attention to each visual token is redundant and costly. Nevertheless, we cannot naively employ only local attention again to reduce complexity, since the generation process differs from image quantization, where global attention maintains the semantic
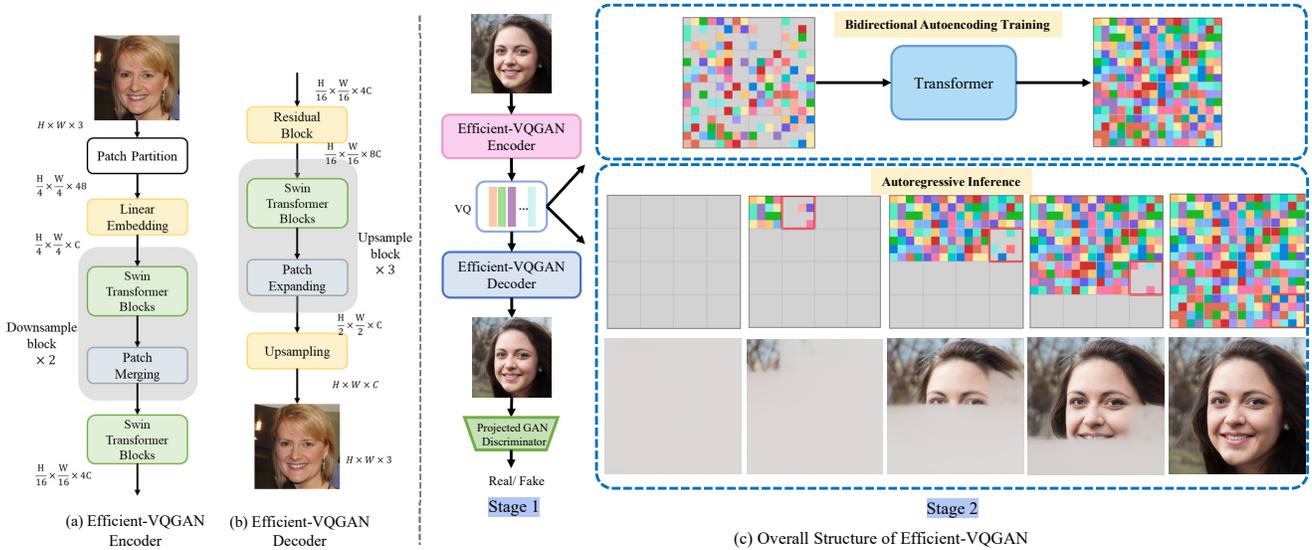
Figure 2. **Overview of Efficient-VQGAN.** Efficient-VQGAN consists of two stages: an encoder-decoder-based vector quantization model (left) and a proposed efficient Transformer model. (c) shows the model architecture and pipeline of Efficient-VQGAN. VQ denotes Vector Quantizer. During the training stage, a subset of tokens is randomly replaced by a mask token (marked in gray) and the Transformer is trained to reconstruct them in a masked autoencoding manner. During the generation stage, starting with masked codes, the model gradually predicts tokens block by block. Within each block, the tokens are iteratively sampled parallelly in a few steps.
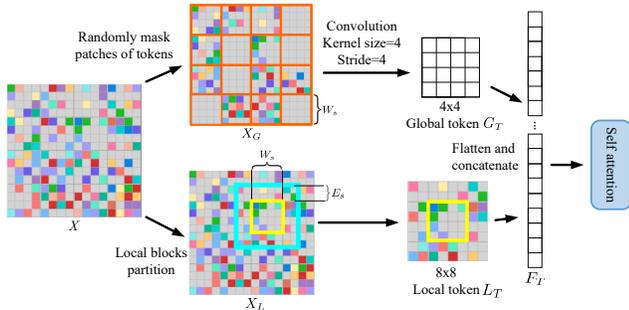

Figure 3. **Multi-Grained Attention calculating process.** Yellow square in X_L denotes the modeled target tokens. Blue square in X_L denotes the extended local token matrix. Each local block will be transformed into a global token (red square in X_G).

consistency of the image. Consequently, both global and local attention is required to model the visual tokens by the transformer.

Inspired by some works [46, 42] which leverages the multi-grained attention mechanism in image recognition task, we combine this mechanism with our masking strategy for image generation. Given an image I, we can obtain the discrete token matrix $X = (x_1, x_2, \cdots, x_{H \times W})$ by passing the image through the Efficient-VQGAN encoder and quantization module, where H, W is the latent size. Then the image token matrix is split into blocks of size $W_s \times W_s$, as shown in Fig. 3. We then perform the multi-grained attention for each query token in the token matrix. Here, we define the query token (the token within yellow square in X_L) as the target tokens will be predicted in this calcula-

tion. Then we define the extend local tokens (blue square in X_L) which is the region surrounding the query block with a block size of $(W_s + 2E_s) \times (W_s + 2E_s)$ to provide more local information cross the blocks, where E_s denotes extend size. As for the global tokens G_T, we separately perform a convolutional operation for each block to group all the tokens in a block into a global token. Here, the filter size and convolutional stride are set to the block size W_s. After this aggregation operation, we obtain $\frac{H}{W_s} \times \frac{W}{W_s}$ global tokens for a token matrix. The final token sequence is defined as the concatenation of global tokens and local tokens, namely $F_T = [G_T, L_T]$. We then feed F_T into Transformer to perform self-attention and output the refined feature that incorporates both the global semantic and local contextual information.

The design of combining block-based global interaction, with local interaction calculated within a local block, leading to considerable reduction in the number of tokens, which facilitate our model to achieve efficient high-resolution image generation.

3.3. Autoencoding Training and Autoregressive Inference

In natural language generation tasks, some researchers [43, 2] point out the pros and cons of BERT-like autoencoding training and autoregressive language modeling. These works aim at exploring a pretraining objective that combines the advantages of both strategies while avoiding their weaknesses. In general, both natural language generation and VQ-based image generation aim at finding

optimal token sequence $s = (s_1, s_2, \cdots, s_l)$. When adopting the autoregressive generation strategy, we can factorize the probability likelihood into a forward product, which can be learned readily by the model $p_\theta(s) = \prod_{i=1}^{l} p_\theta^i(s_i | s_{<i})$. Nonetheless, autoregressive models are challenging to capture deep bidirectional context information due to the uni-directional training constraint. On the other hand, BERT-like bidirectional autoencoding training strategy allows the model to capture bidirectional context better, however, an independent assumption is required that the tokens predicted in parallel ought to be independent of each other, otherwise semantic inconsistency will occur. Accordingly, in this section, we propose our training and inference pipeline, which incorporates the unique advantages of autoencoding training and autoregressive generation.

Masked Autoencoding training Strategy. Given the discrete token sequence $X = (x_1, x_2, \cdots, x_{H \times W})$, following MaskGIT [5], a subset of tokens are randomly masked. The mask ratio is defined by a cosine scheduling function $\gamma(r) = \text{cosine}(\frac{\pi}{2} r) \in (0, 1]$, where the ratio r is from 0 to 1. We uniformly choose $\lceil \gamma(r) \cdot (H \times W) \rceil$ tokens in X and replace them with mask token, producing the corrupted token matrix $X_L = (x_1, [\text{MASK}], \cdots, [\text{MASK}], x_{H \times W})$.

During the training process, we divide the masking strategy into two parts, for calculating local and global attention, respectively. Local attention is performed on X_L within each block as mentioned above to produce L_T. When calculating global attention, entire patches of tokens will be randomly masked first. This operation enables our model to predict blocks in arbitrary order, including autoregressive order and non-autoregressive order, which also make it possible to achieve inpainting, outpainting and image editing tasks. Then we perform a convolutional operation that transforms each block in X_G into one global token, producing a corrupted global token matrix with $\frac{H}{W_s} \times \frac{W}{W_s}$ global visual tokens, namely G_T. After obtaining all tokens F_T, we feed them to the Transformer module to predict the probability distribution of each masked token within one block. The loss function can be formulated as:

$$\mathcal{L} = -\mathbb{E} \left[\sum_{\forall i \in [1, H \times W], L_{T_i} = [\text{MASK}]} \log p(x_i | L_T, G_T) \right], \quad (2)$$

where the negative log-likelihood is computed as the cross-entropy reconstruction loss between the true one-hot token within a block and the predicted token.

Autoregressive Inference Strategy. In previous autoregressive decoding methods [11, 44], tokens are sequentially generated based on all previously generated tokens, which could improve the generation consistency. For high-resolution image generation, however, the sampling speed is intolerable due to the long sequence length. Based on bidirectional training, MaskGIT [5] can generate multiple
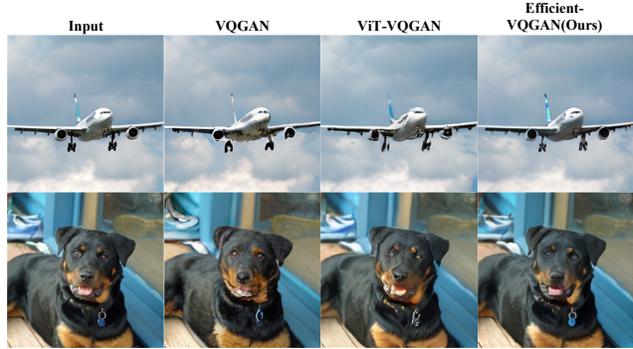

Figure 4. **Reconstruction comparison between VQGAN [11], ViT-VQGAN [44] and Efficient-VQGAN on ImageNet dataset.** Ours can perfectly reconstruct the original image, preserving more details compared to others.

image tokens in a single pass, and iteratively generate a complete image, which greatly reduces the sampling steps during inference. However, as claimed in [38], when sampling multiple tokens simultaneously and each token is sampled independently with an estimated probability will result in ignoring the dependencies between different tokens at different locations. In [38], they propose a fewer tokens sampling strategy that samples fewer tokens at each step to alleviate this joint distribution issue. Inspired by such strategy, we propose to generate image tokens block by block in an autoregressive manner, which incorporates the design of our block-based model architecture and the merit of autoregressive sampling. The difference compared to [38] is that [38] fixes the number of tokens generated in the whole image for each step, we fix the order of generation for each step, which essentially achieves a similar effect. Besides, in each block, we adopt parallel decoding used in MaskGIT that generates multiple tokens at the same time within a block. Under the setting of MaskGIT training strategy that randomly masks tokens in a whole image, it is difficult to include all the situations during inference stage (different masked token mode), especially for high-resolution image generation. However, limiting parallel sampling to a single block in our inference strategy would greatly alleviate this problem. Therefore, our sampling strategy follows an overall autoregressive and local parallel generation manner.

4. Experiments

In this section, we evaluate the ability of Efficient-VQGAN in the first vector quantization stage (Sec. 4.1) and image synthesis tasks (Sec. 4.2) on FFHQ [21], CelebA-HQ [20], and ImageNet [7] dataset. In Sec. 4.3, we present some direct applications of Efficient-VQGAN on image inpainting, outpainting, and editing tasks. In Sec. 4.4, we conduct some ablation studies. The following tests are implemented on 256^2 images unless otherwise specified. Details of experiment settings can be found in our appendix.

Figure 5. **Synthesized samples by Efficient-VQGAN on ImageNet dataset at 256^2 resolution.**

Methods	Dataset	Codebook Size	Latent Size	FID on Validation ↓
VQGAN [11]	FFHQ	1024	16x16	4.39
ViT-VQGAN [44]	FFHQ	8192	32x32	3.13
Ours	FFHQ	1024	16x16	3.38
Ours	FFHQ	1024	32x32	**2.72**
VQGAN	ImageNet	16384	16x16	4.98
VQGAN*	ImageNet	8192	32x32	1.49
VQGAN**	ImageNet	512	64x64 & 32x32	1.45
ViT-VQGAN	ImageNet	8192	32x32	1.55
Ours	ImageNet	1024	16x16	2.34
Ours	ImageNet	1024	32x32	**0.95**

Table 1. **Fréchet Inception Distance (FID) [14] between reconstructed validation split and original validation split.** ∗ means model trained with Gumbel-Softmax strategy. ∗∗ means model leveraging multi-scale hierarchical codebook proposed in [17]. Ours shows the best reconstruction quality.

Methods	Downsampling factor f	Latent Size	Throughput (imgs/sec) ↑	FID↓
VQGAN [11]	16	16x16	112	4.98
Ours	16	16x16	**142**	**2.34**
VQGAN [11]	8	32x32	103	1.49
ViT-VQGAN [44]	8	32x32	95	1.55
Ours	8	32x32	**108**	**0.95**

Table 2. **Reconstruction speed comparison**. Under the same latent size, ours achieve faster reconstruction speed and better quality. Tests are implemented on single A100-80GB GPU.

4.1. Image Quantization

When training the image quantization model, we follow the default train and validation split for each dataset and conduct experiments with different downsampling factor f. For each setting, the Codebook Size $|Z|$ is set to 1024. On FFHQ and CelebA-HQ dataset, our models are trained with a batch size of 16 in 8*A100 GPU for a total of 50 epochs, while on ImageNet we find that the model has converged for only 20 epochs. Quantitative reconstruction comparison results (256^2 resolution) are shown in Tab. 1 and Tab. 2. Our method achieves higher inference speed due to the re-

Figure 6. **Synthesized samples by Efficient-VQGAN on FFHQ dataset at 1024^2 resolution.** Best viewed zoomed in.

duced computation cost of local attention operations. Besides, under the same downsampling factor f, our model achieves better reconstruction fidelity with a lower capacity of codebook (1024 codebook entries). Qualitative comparison results display in Fig. 4. We claim that the global attention mechanism may bring some noises during image quantization so that our method which only performs local attention could put more computation on local interactions and improve generative details. More quantitative results for different resolutions ($512^2, 1024^2$) can be found in our appendix.

4.2. Image Synthesis

Based on the well-trained Efficient-VQGAN autoencoder, we train our transformer model and evaluate the performance on unconditional and class-conditioned image synthesis tasks. To speed up the generation process, we set the downsampling factor f of the first stage to 16. Our models are trained in 8*A100 GPU for a total of 200 epochs. When synthesizing samples, in-block iteration steps $T = 8$

CelebA-HQ 256 × 256		FFHQ 256 × 256	
Methods	FID ↓	Methods	FID ↓
NVAE [39]	40.3	BigGAN [3]	12.4
VAEBM [41]	20.4	ImageBART [10]	9.57
Style ALAE [30]	19.2	GANformer [16]	7.42
DC-VAE [28]	15.8	VQ-Diffusion [13]	6.33
StyleSwin [45]	3.25	StyleGAN-XL [37]	2.19
VQGAN [11]	10.2	VQGAN [11]	9.6
VIM-Large [44]	7.0	VIM-Large [44]	5.3
Ours	7.81	Ours	5.28

Table 3. **Quantitative comparison of face image generation.**

Methods	FID↓
StyleGAN-XL [37]	2.02
StyleGAN2 [22]	2.84
HiT-B [47]	6.37
Ours	11.81
StyleALAE [30]	13.09

Table 4. **Quantitative comparison on FFHQ at 1024^2 resolution.**

is applied.

We compare the quantitative results of our Efficient-VQGAN with several state-of-the-art methods on FID score. The results of unconditional image synthesis on CelebA-HQ and FFHQ are shown in Tab. 3. While some task-specialized GAN models report better FID scores, our model architecture is more flexible and can support a variety of tasks. The reported FID score of VIM-large [44] on CelebA-HQ is slightly better than ours, due to the number of parameters that are 8 times larger than our model. Quantitative comparisons of face synthesis at 1024 resolution are shown in Tab. 4. Existing methods are almost GAN-based, and to our knowledge, no VQ-based method reports better performance. Compared to our baseline VQ-GAN [11] and VIM-base [44], we also improve the quality of class-conditioned image synthesis as shown in Tab. 5. The impressive generated images of ImageNet dataset are shown in Fig. 5. Furthermore, we evaluate the memory cost of existing vector quantized transformer models during training time, as shown in Tab. 6. When the image token sequence is too long, the out-of-memory(OOM) issue occurs in other methods due to the computational overhead of the global self-attention mechanism. Benefiting from the multi-grained attention design, our model requires less memory resources and can synthesize higher resolution images (1024^2), as shown in Fig. 6. We also adopt the pretraind VQGAN quantizer [11] as the first stage of our model and retrained the generative model for the second stage. On the basis of the same quantization model, the generation results of our model greatly exceed the baseline VQGAN model [11] (see Tab. 7), demonstrating the effectiveness of our multi-grained attention and overall generation pipeline.

Methods	Acceptance Rate	FID ↓	IS ↑
IDDPM [27]	1	12.3	-
StyleGAN-XL [37]	1	2.3	265.12
VQ-Diffusion [13]	1	11.89	-
ADM-G [8]	1	10.94	101
VIM-Base [44]	1	11.2	97.2
MaskGIT [5]	1	6.18	182.1
VQGAN [11]	1	17.04	70.6
VQGAN [11]	0.5	10.26	125.5
Ours	1	9.92	82.2
Ours	0.5	6.81	135.84

Table 5. **FID comparison of class-conditioned image synthesis on ImageNet at 256^2 resolution.** All VQ-based models above take 16×16 latent size. Acceptance rate reports classifier-based rejection sampling using ResNet-101.

Methods	#Params	Training Time Memory Cost [M] ↓		
		Latent Size 32x32	Latent Size 64x64	Latent Size 128x128
VQGAN [11]	1.4B	21848	OOM	OOM
VIM-Base [44]	650M	11638	OOM	OOM
MaskGIT [5]	227M	6780	49056	OOM
Ours	185M	5560	14154	58096

Table 6. **Training time memory cost of Stage-2 models in different latent size.** For a fixed image quantization compress rate, a larger latent size indicates higher image resolution. OOM denotes out-of-memory.

Dataset	Ours FID ↓	Baseline FID [11] ↓
FFHQ	7.5	9.6
ImageNet	13.67	17.04

Table 7. **FID comparison of second stage model between Ours and VQGAN based on the same first stage quantization model.** Proposed model greatly outperforms VQGAN stage-2 baseline.

4.3. Image Editing Applications

Due to causality limitations in the inference process, it is challenging for autoregressive generative models to perform image editing. Efficient-VQGAN can be seamlessly applied to three image editing tasks (see Fig. 7), without any modifications of the model architecture. We tokenize the input masked image and feed the corrupted image token matrix into the second stage to iteratively complete the masked image. As shown in the first and second row of Fig. 7, Efficient-VQGAN can make consistent completions thanks to the multi-grained interaction that allows the model to capture both the global semantic as well as the local fine-grained details. Further, due to the randomness in the inference stage, we can obtain diverse editing results. Class-conditional image editing is defined as regenerating the image content inside a bounding box conditioned on the given class label. Efficient-VQGAN can replace the selected object while preserving the background outside the bounding box, and the entire image is visually harmonious.
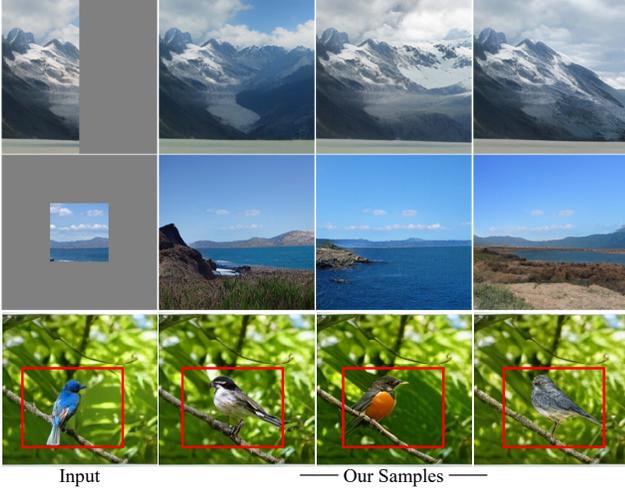
Figure 7. **Inpainting, outpainting and class-conditional image editing.** While maintaining the semantic consistency of images, our model shows the diversity of inpainting (first row) and outpainting (second row). Replace the given image in the bounding box with a target class object (third row).

4.4. Ablation Studies

We conduct ablation studies at 256^2 resolution images with 16×16 latent size. More ablation results could be found in our appendix.

Block Size W_s. We evaluate the generation performance with the block size of $2, 4, 8, 16$ in the left image of Fig. 8. When the block size is too small, *e.g.*, $2 * 2$, the model is prone to overfitting since there is only $2^4 = 16$ masked mode when learning local relationship within a block. It's worth noting that when block size is equal to latent size, i.e. $W_s = 16$ here, our model will degrade into MaskGIT [5], that is, there is only one block during inference. Then all tokens will be predicted in a non-autoregressive manner. As we mentioned in Sec. 3.3, due to the large number of different masked patterns, the training phase cannot cover all the cases in inference stage. Besides, predicting many tokens simultaneously will lead to a more serious joint distribution problem, resulting in a bad FID score. The block size of 8 obtains the best FID score since the suitable block size allows the model to learn both global semantics and local details relatively well. This comparison results shows the advantage of our sampling method over MaskGIT. Besides, to study the effect of block size on generation speed, we compare autoregressive method, *i.e.*, VQGAN and our model with different block sizes. As shown in Fig. 9, when $W_s = 1$, our model performs as an autoregressive method. Ours with a larger block size runs faster, demonstrating the efficiency of the block-based design.

In-block Iteration Steps T. We investigate the influence of the number of iterations within a block for image synthesis, as shown in the right image of Fig. 8. We assess the generated images as $T = 4, 8, 12, 16, 24, 32$ on two models with
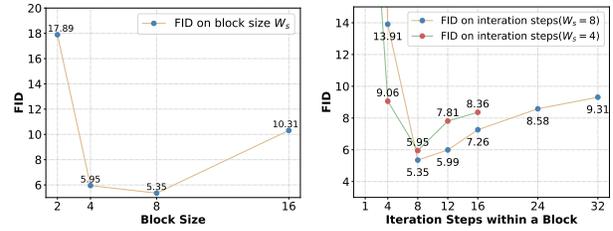

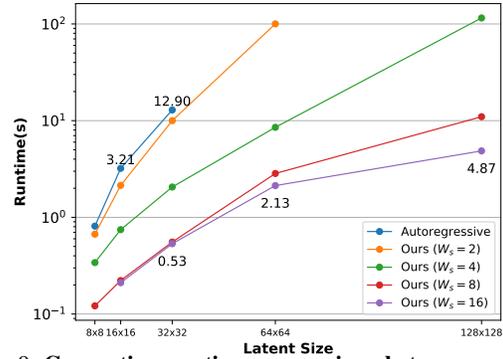Figure 8. **Ablation study on the block size and in-block iteration steps.**

Figure 9. **Generation runtime comparison between ours with different block size W_s and autoregressive method [11].**

different block sizes. When T increases from 4 to 8, the FID gradually decreases since fewer steps means more tokens will be predicted simultaneously, which is hard to meet the independence assumption. However, more iteration steps do not lead to better quality because more iteration steps means more predictions based on known tokens, which narrows the sample probability distribution across the dataset, leading to less diversity. This observation is consistent with the findings in MaskGIT [5].

5. Conclusion and Discussion

This paper proposes Efficient-VQGAN, an efficient two-stage vector quantized model, for high-resolution image generation. We make several improvements in both the first quantization and second generative modeling stage, contributing to higher computational efficiency and generation quality. A new image generation paradigm is developed that combines the masked autoencoding training and autoregressive inference, facilitating better generation quality. As for the limitations, we sample multiple tokens in parallel within each block, increasing the sampling speed at the cost of slightly reducing the quality of generated images. Devising a better inference strategy to combine multi-grained attention can be interesting future work.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022.

[2] Bin Bi, Chenliang Li, Chen Wu, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. Palm: Pre-training an autoencoding&autoregressive language model for context-conditioned generation. *EMNLP*, 2020.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *ICLR*, 2018.

[4] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. *arXiv preprint arXiv:2105.05537*, 2021.

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

[6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pre-training from pixels. In *ICML*, 2020.

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

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

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[10] Patrick Esser, Robin Rombach, Andreas Blattmann, and Bjorn Ommer. Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. *NeurIPS*, 2021.

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

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 27, 2014.

[13] 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.

[14] 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. *NeurIPS*, 2017.

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

[16] Drew A Hudson and Larry Zitnick. Generative adversarial transformers. In *ICML*, 2021.

[17] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[18] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two pure transformers can make one strong gan, and that can scale up. *Advances in Neural Information Processing Systems*, 34:14745–14758, 2021.

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, 2016.

[20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020.

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

[24] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022.

[25] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *CVPR*, 2021.

[26] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *CVPR*, 2021.

[27] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021.

[28] Gaurav Parmar, Dacheng Li, Kwonjoon Lee, and Zhuowen Tu. Dual contradistinctive generative autoencoder. In *CVPR*, 2021.

[29] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *ICML*, 2018.

[30] Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *CVPR*, 2020.

[31] 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.

[32] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

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

[34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

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

[36] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. *NIPS*, 2021.

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

[38] Zhicong Tang, Shuyang Gu, Jianmin Bao, Dong Chen, and Fang Wen. Improved vector quantized diffusion models. *arXiv preprint arXiv:2205.16007*, 2022.

[39] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *NeurIPS*, 2020.

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

[41] Zhisheng Xiao, Karsten Kreis, Jan Kautz, and Arash Vahdat. Vaebm: A symbiosis between variational autoencoders and energy-based models. *ICLR*, 2020.

[42] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *arXiv preprint arXiv:2107.00641*, 2021.

[43] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *NeurIPS*, 2019.

[44] 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. *ICLR*, 2021.

[45] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *CVPR*, 2022.

[46] Pengchuan Zhang, Xiyang Dai, Jianwei Yang, Bin Xiao, Lu Yuan, Lei Zhang, and Jianfeng Gao. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In *ICCV*, 2021.

[47] Long Zhao, Zizhao Zhang, Ting Chen, Dimitris Metaxas, and Han Zhang. Improved transformer for high-resolution gans. *Advances in Neural Information Processing Systems*, 34:18367–18380, 2021.

[48] Shengjia Zhao, Hongyu Ren, Arianna Yuan, Jiaming Song, Noah Goodman, and Stefano Ermon. Bias and generalization in deep generative models: An empirical study. *NeurIPS*, 2018.