MaskGIT

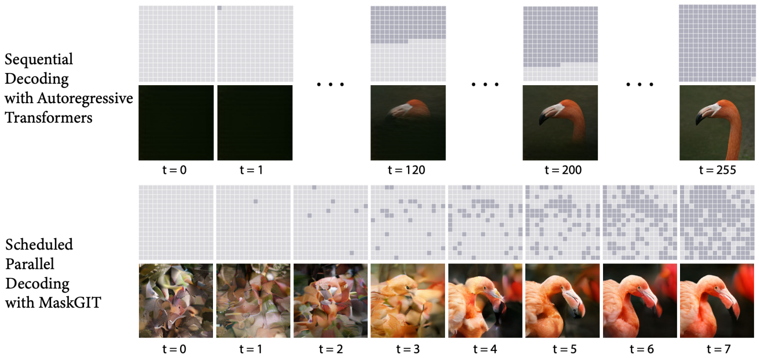

Figure 2. **Comparison between sequential decoding and MaskGIT's scheduled parallel decoding.** Rows 1 and 3 are the input latent masks at each iteration, and rows 2 and 4 are samples generated by each model at that iteration. Our decoding starts with all unknown codes (marked in lighter gray), and gradually fills up the latent representation with more and more scattered predictions in parallel (marked in darker gray), where the number of predicted tokens increases sharply over iterations. MaskGIT finishes its decoding in 8 iterations compared to the 256 rounds the sequential method takes.

- □□□□□□MaskGIT: Masked Generative Image Transformer
- □□□□□□https://arxiv.org/abs/2202.04200
- CVPR 2022

□□□□□□MaskGIT□□□□□□□□□□□□□Transformer□□□□□□□□□□□token□□□□□□□□□□□□□□token□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□SOTA□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□GANs□□□Transformer□NLP□□□□□□Transformer□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□token□□MaskGIT□□□□□□□□□□□□□□□□transformer□

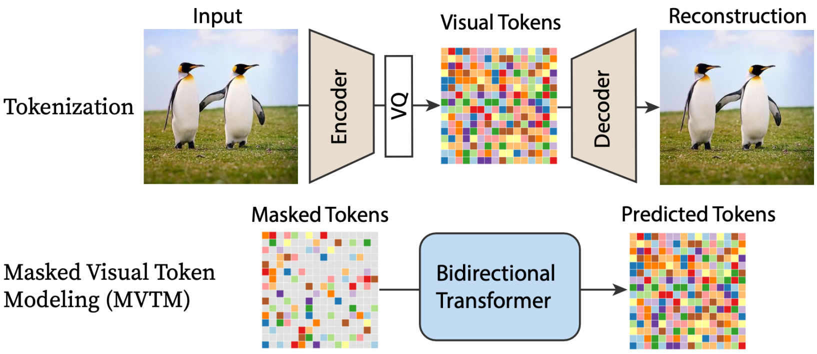

Figure 3. **Pipeline Overview.** MaskGIT follows a two-stage design, with 1) a tokenizer that tokenizes images into visual tokens, and 2) a bidirectional tranformer model that performs MVTM, i.e. learns to predict visual tokens masked at random.

□□□□□□□□□□□□□□□□□□□□□□□□□□□tokenization□□□□□□□□□□□□□□□□□□□□□□□□□□□□token□□□□□□□□□□□□□□□□□□□□□□□□□□□□□*Masked Visual Token Modeling* (MVTM)□□□□□□□□□□□□transformer□

□□□□□□□□□□□□□□□□□□□□□□□□□token□□□□□□0-1□□□token□□mask□□□□□□mask□□token□□mask□□token□□□□□□□□□□□□□□□□

$$\mathcal{L}_{\text{mask}} = - \mathop{\mathbb{E}}_{\mathbf{Y}\in\mathcal{D}} \left[\sum_{\forall i \in [1,N], m_i = 1} \log p(y_i | Y_{\overline{\mathbf{M}}}) \right]$$

mask□□□□□□□□□□□□transformer□□□□□□□□□□□□□□□□□□□□□□□□□transformer□□□□□□□□□□□□□□□□□□□□token□□□□□token□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□token□□□□□□□□□□□□□□□□□□□□□□□mask□□□□□□□□□□mask□token□□□□□□□□□□□□□□□□token□□mask□□□□□□□□□token□mask□□□□□□□□□□□□MaskGIT□decode□□□□□□□□mask□□□□□□□□□□□□□□□□mask□token□□

γ	T	FID ↓	IS ↑	NLL

Exponential	8	7.89	156.3	4.83
Cubic	9	7.26	165.2	4.63
Square	10	6.35	179.9	4.38
Cosine	10	**6.06**	**181.5**	4.22
Linear	16	7.51	113.2	3.75
Square Root	32	12.33	99.0	3.34
Logarithmic	60	29.17	47.9	3.08

Table 3. **Ablation results on the mask scheduling functions.** We report the best FID, IS, and Negative Log-Likelihood loss for each candidate scheduling function.

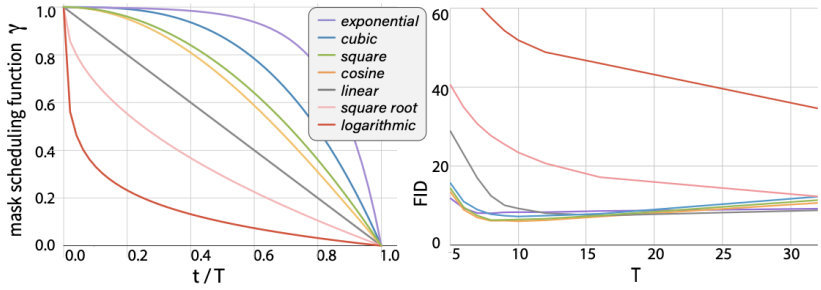

Figure 8. **Choices of Mask Scheduling Functions** $\gamma(\frac{t}{T})$, and **number of iterations** $\boldsymbol{T}$. On the left, we visualize seven functions we consider for γ. On the right, we show line graphs of models' FID scores against the number of decoding iterations T. Among the candidates, we find that cosine achieves the best FID.

□□□□□□□□□□□□□□□□□□□□mask□□□
□□

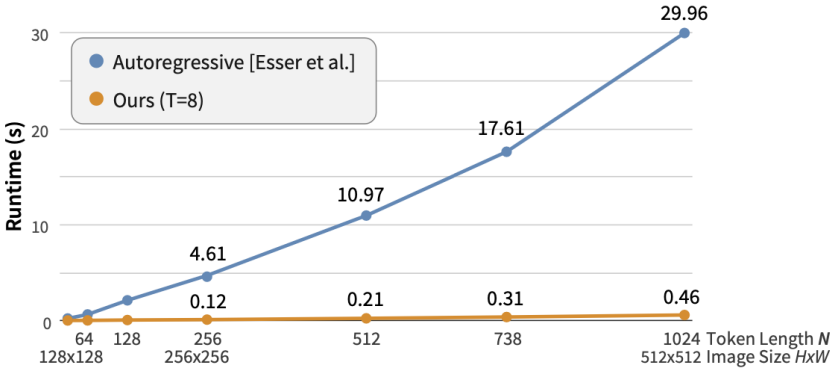

Figure 4. **Transformer wall-clock runtime comparison between VQGAN [15] and ours.** All results are run on a single GPU.

Model	FID ↓	IS ↑	Prec ↑	Rec ↑	# params	# steps	CAS ×100 ↑	
							Top-1 (76.6)	Top-5 (93.1)
ImageNet 256×256								
DCTransformer [32]□	36.51	n/a	0.36	**0.67**	738M	>1024		
BigGAN-deep [4]	6.95	**198.2**	**0.87**	0.28	160M	1	43.99	67.89
Improved DDPM [33]□	12.26	n/a	0.70	0.62	280M	250		
ADM [12]□	10.94	101.0	0.69	0.63	554M	250		
VQVAE-2 [37]□	31.11	∼45	0.36	0.57	13.5B†	5120	54.83	77.59
VQGAN [15]□	15.78	78.3	n/a	n/a	1.4B	256		
VQGAN*	18.65	80.4	0.78	0.26	227M	256	53.10	76.18
MaskGIT (Ours)	**6.18**	182.1	0.80	0.51	227M	*8*	**63.14**	**84.45**
ImageNet 512×512								
BigGAN-deep [4]	8.43	**232.5**	**0.88**	0.29	160M	1	44.02	68.22
ADM [12]□	23.24	58.06	0.73	**0.60**	559M	250		
VQGAN*	26.52	66.8	0.73	0.31	227M	1024	51.29	74.24
MaskGIT (Ours)	**7.32**	156.0	0.78	0.50	227M	*12*	**63.43**	**84.79**

Table 1. Quantitative comparison with state-of-the-art generative models on ImageNet 256×256 and 512×512. "# steps" refers to the number of neural network runs needed to generate a sample. * denotes the model we train with the same architecture and setup with ours; □ denotes values taken from prior publications; † estimated based on the pytorch implementation [39].

Task	Model	FID ↓	IS ↑
Outpainting	Boundless [43]□	35.02	6.15

Outpainting	Boundless [43]	33.02	6.15
Right 50%	In&Out [8]$^{\square}$	23.57	7.18
	InfinityGAN [31]	10.60	5.57
	Boundless [43] TF $^{\blacklozenge}$	7.80	5.99
	MaskGIT (Ours) ^{512}	**6.78**	**11.69**
Inpainting	DeepFill [52]	11.51	22.55
Center 50%×50%	ICT [49]†	13.63	17.70
	HiFill [50]^{512}	16.60	19.93
	CoModGAN [57]^{512}	**7.13**	21.82
	MaskGIT (Ours)^{512}	7.92	**22.95**

Table 2. **Quantitative Comparisons for Inpainting and Outpainting on Places2.** ^{512} evaluated on 512×512 samples while others evaluated on the corresponding 256×256 ones, consistent with their training; $^{\square}$ taken from the prior work; † evaluated using the released model trained on a subset of Places2; $^{\blacklozenge}$ evaluated using the TFHub model [18].

Newer Older

2024-07-20
OpenVoice

2024-07-03
Spectron