Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, Liwei Wang

STRUCT Group Seminar
Presenter: Haowei Kuang
2024.12.08

NeurIPS 2024 Best Paper

OUTLINE

- Authorship
- Background
- Method
- Experiments
- Conclusion

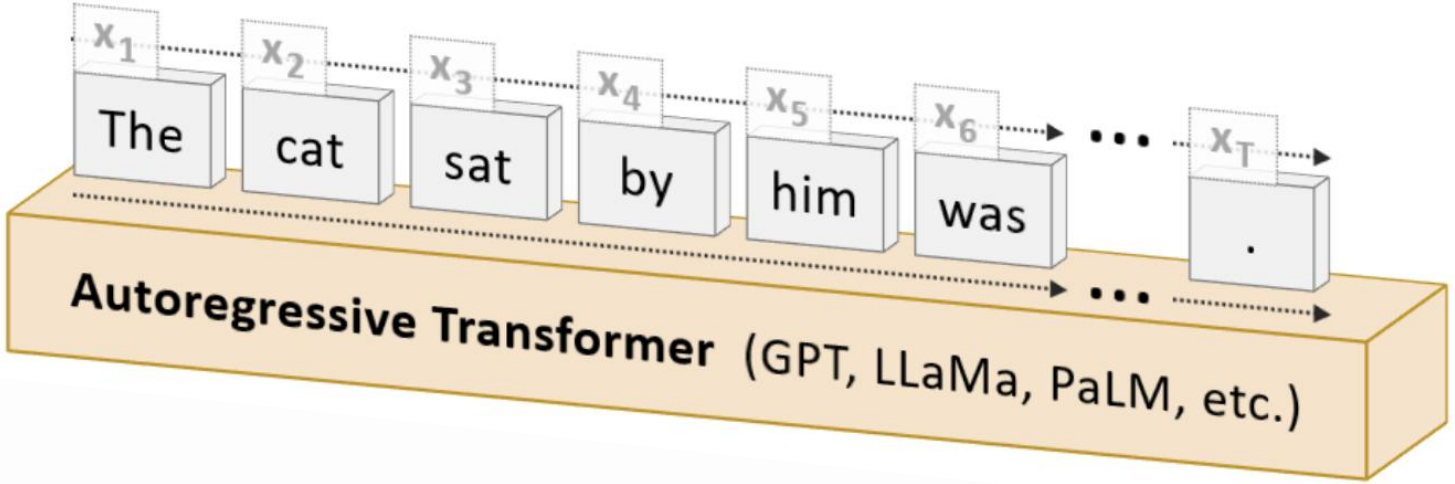
OUTLINE

- Authorship
- Background
- Method
- Experiments
- Conclusion

BACKGROUND: Autoregressive Model (AR)

Autoregressive Model
posits the probability of current token x_T depends only on its prefix

$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t \mid x_1, x_2, \ldots, x_{t-1}).$$



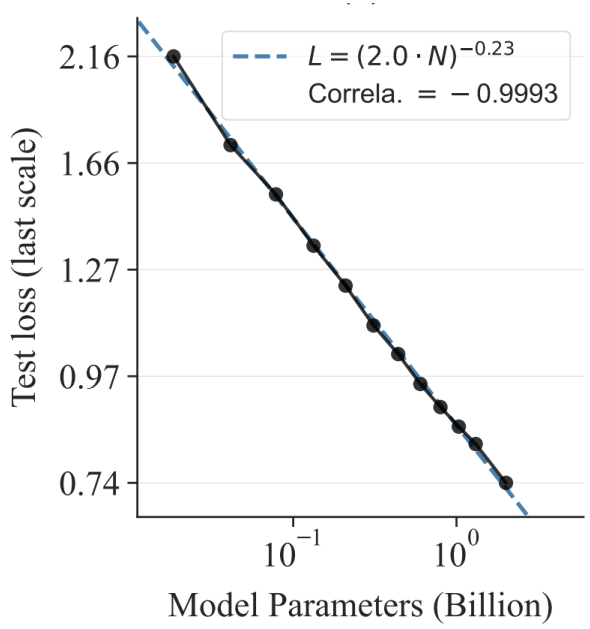
BACKGROUND: Autoregressive Model (AR)

Scaling law

The performance correlates with parameters and optimal training compute, following a power-law:

$$L = (\beta \cdot X)^{\alpha}$$

$$\log(L) = \alpha \log(X) + \alpha \log \beta$$



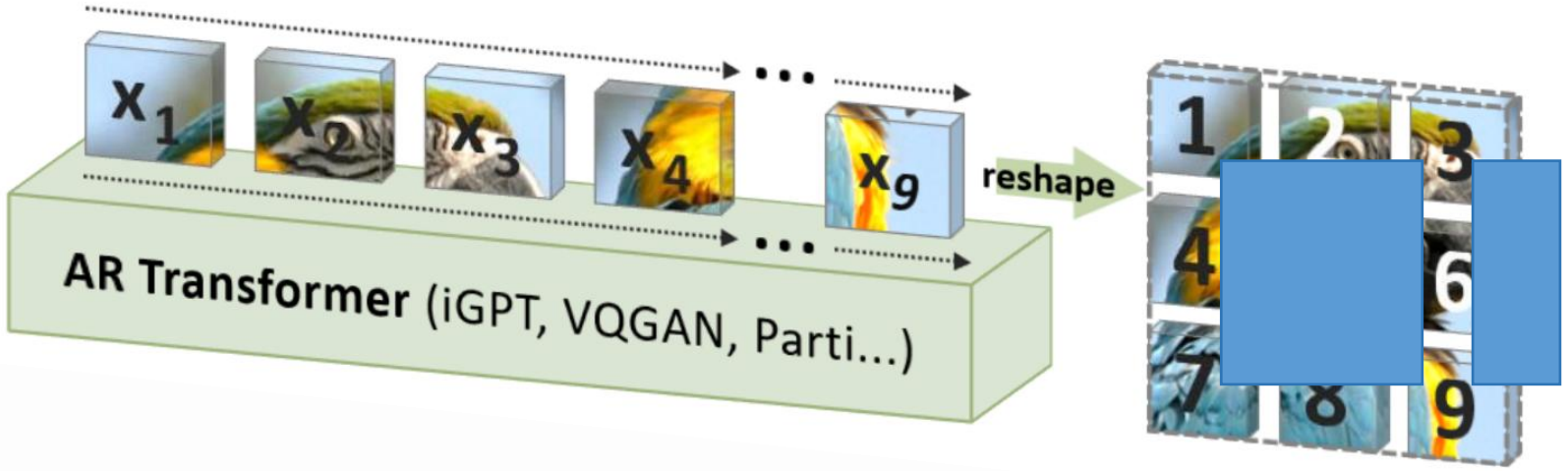
Zero-shot generalization

LLM can perform tasks that it has not been explicitly trained on.

BACKGROUND: Autoregressive Model (AR)

Autoregressive Model on Visual Generation

- Tokenize an image into several discrete tokens
- Define a 1D order of tokens for unidirectional modeling
- Training an autoregressive model for unidirectional modeling



BACKGROUND: Autoregressive Model (AR)

Autoregressive Model on Visual Generation

- Tokenize an image into several discrete tokens

With a quantized autoencoder to convert the image to discrete tokens

$$f = \mathcal{E}(im), \qquad q = \mathcal{Q}(f)$$

$$q^{(i,j)} = \left(\underset{v \in [V]}{\arg\min} \|\text{lookup}(Z, v) - f^{(i,j)}\|_2 \right) \in [V]$$

$$f \in \mathbb{R}^{h \times w \times C}$$
$$Z \in \mathbb{R}^{V \times C}$$

$$\hat{f} = \text{lookup}(Z, q), \qquad \hat{im} = \mathcal{D}(\hat{f})$$

Training Loss: $\mathcal{L} = \|im - \hat{im}\|_2 + \|f - \hat{f}\|_2 + \lambda_P \mathcal{L}_P(\hat{im}) + \lambda_G \mathcal{L}_G(\hat{im})$

BACKGROUND: Autoregressive Model (AR)

Autoregressive Model on Visual Generation

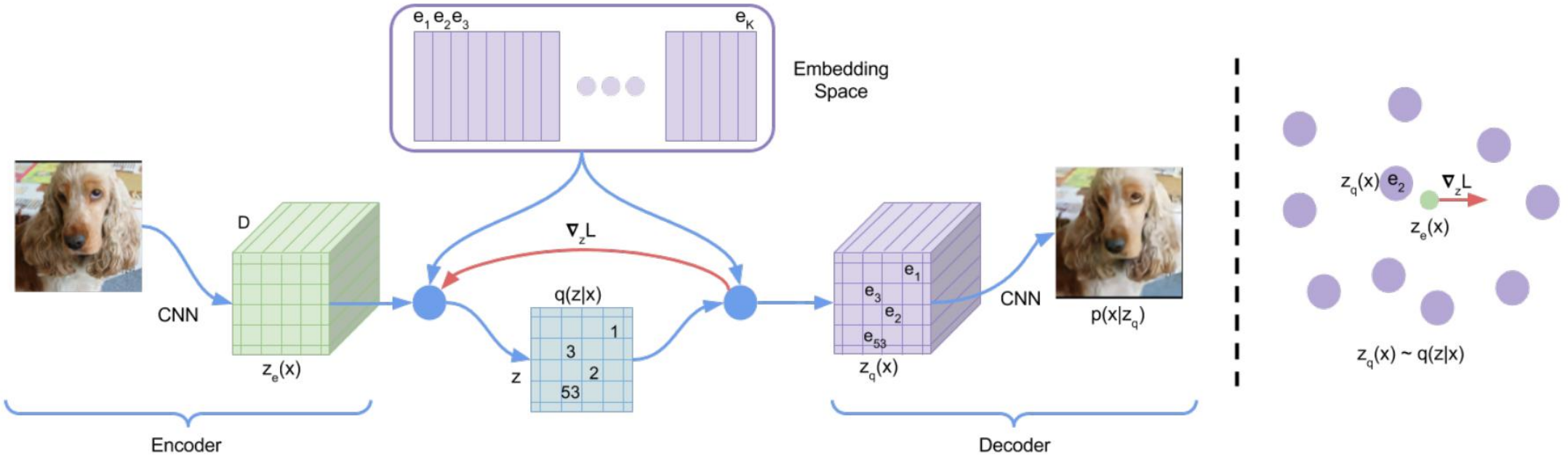
- Tokenize an image into several discrete tokens
- Define a 1D order of tokens for unidirectional modeling
- Training an autoregressive model for unidirectional modeling

$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t \mid x_1, x_2, \ldots, x_{t-1}).$$

BACKGROUND: AR on Visual Generation

VQ-VAE

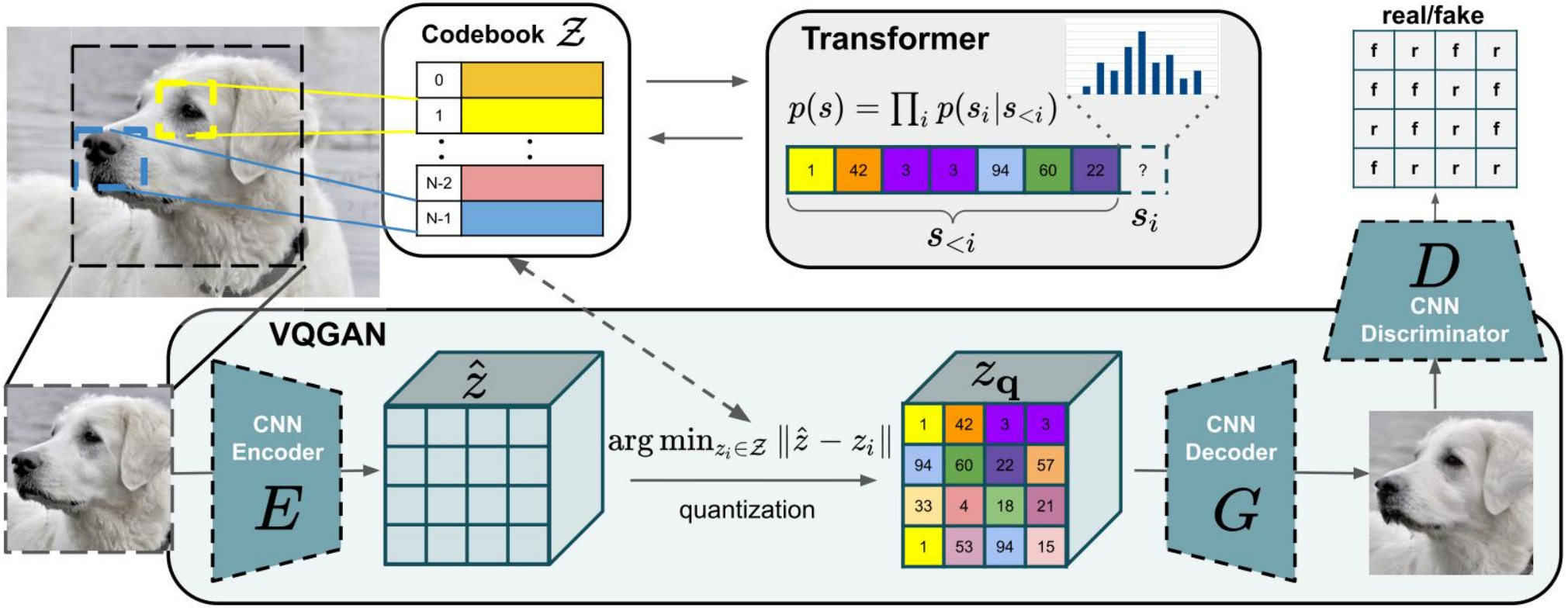
- Introduce Vector Quantisation into VAE



BACKGROUND: AR on Visual Generation

VQ-GAN

- Introduce GAN Loss and Transformer Structure



BACKGROUND: AR on Visual Generation

Weakness: Autoregressive Model on Visual Generation

- Mathematical premise violation

  the token sequence $(x_1, x_2, \ldots, x_{h \times w})$ retains bidirectional correlations, contradicting the unidirectional dependency assumption.

- Inability to perform some zero-shot generalization

  The unidirectional nature of image autoregressive modeling restricts their generalizability in tasks requiring bidirectional reasoning.

BACKGROUND: AR on Visual Generation

Weakness: Autoregressive Model on Visual Generation

- Structural degradation
  The flattening disrupts the spatial locality inherent in image feature maps.

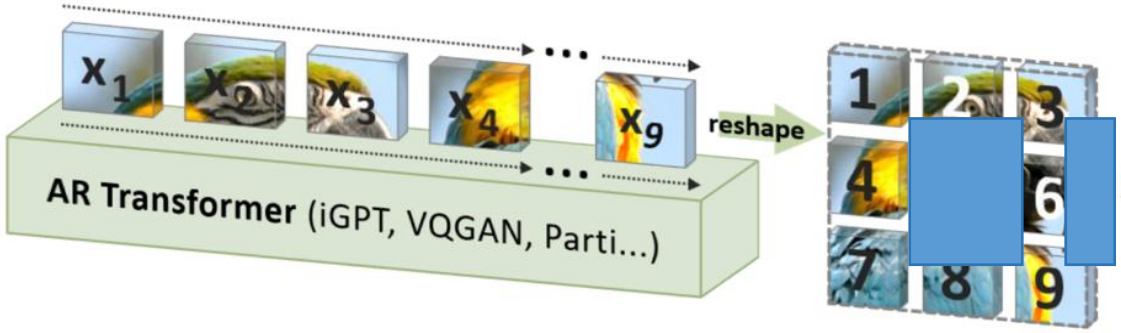
- Inefficiency
  A conventional self-attention transformer incurs O(n^2) autoregressive steps and O(n^6) computational cost.

OUTLINE

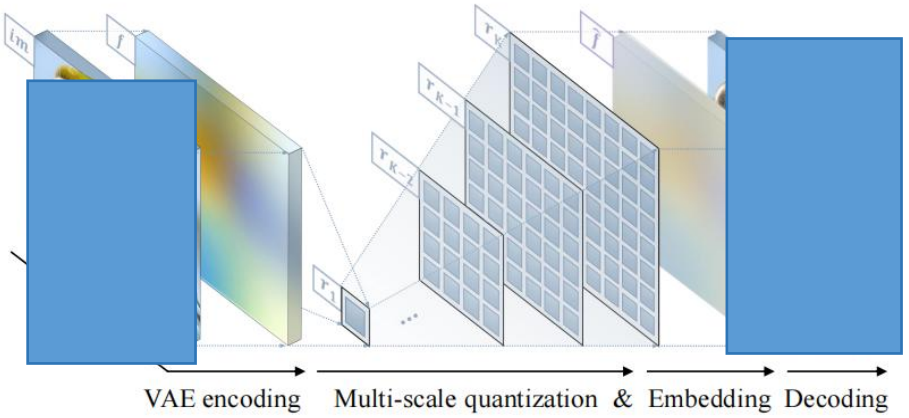
- Authorship
- Background
- Method
- Experiments
- Conclusion

METHOD: Visual Autoregressive Modeling (VAR)

Shifting from “next token prediction” to “next-scale prediction” strategy



$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t \mid x_1, x_2, \ldots, x_{t-1})$$



$$p(r_1, r_2, \ldots, r_K) = \prod_{k=1}^{K} p(r_k \mid r_1, r_2, \ldots, r_{k-1})$$

METHOD: Visual Autoregressive Modeling (VAR)

Visual Autoregressive Model on Visual Generation

- Mathematical premise violation

  New coarse-to-fine constraint is acceptable as it aligns with the natural

- Structural degradation

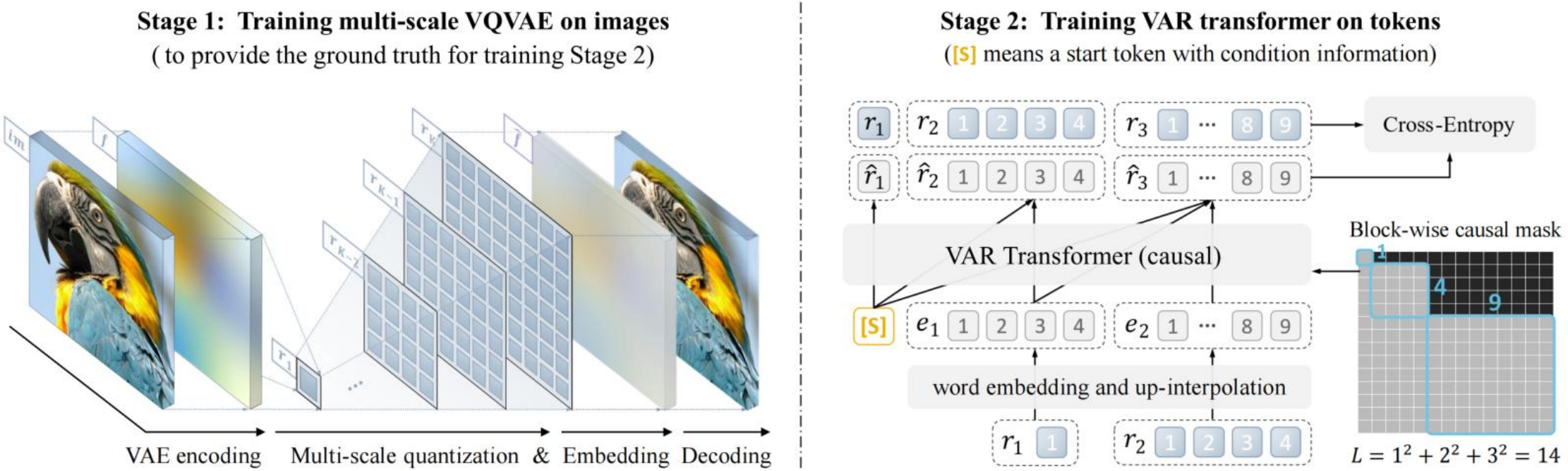
  (i) there is no flattening operation in VAR (ii) tokens in each r_k are fully correlated

- Inefficiency

  O(k) autoregressive steps and O(n^4) computational cost

METHOD: Visual Autoregressive Modeling (VAR)

Two Stage Training



METHOD: Visual Autoregressive Modeling (VAR)

Tokenization

Algorithm 1: Multi-scale VQVAE Encoding

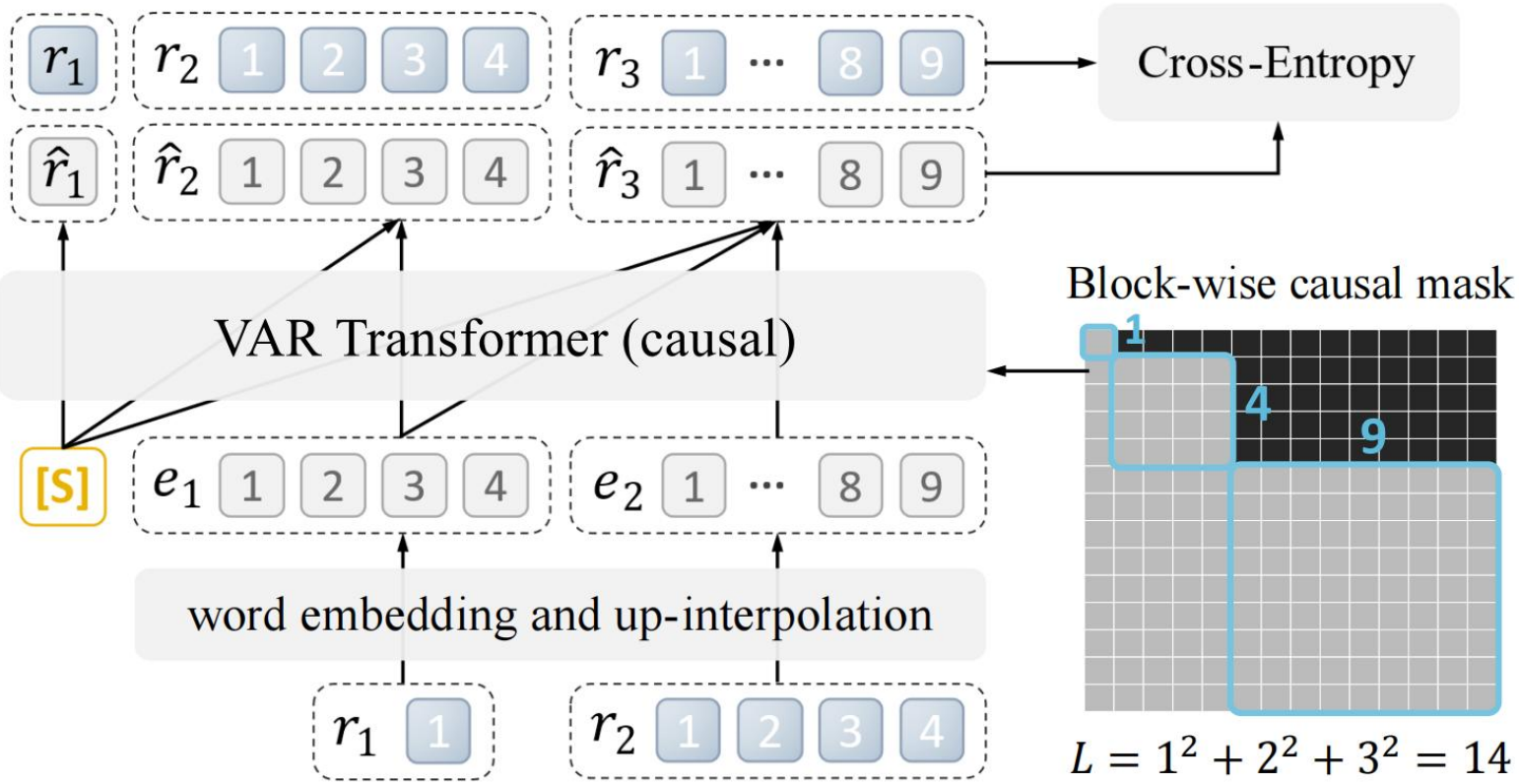
Inputs: raw image im;
Hyperparameters: steps K, resolutions $(h_k, w_k)_{k=1}^{K}$;
$f = \mathcal{E}(im)$, $R = []$;
for $k = 1, \cdots, K$ **do**
    $r_k = \mathcal{Q}(\text{interpolate}(f, h_k, w_k))$;
    $R = \text{queue\_push}(R, r_k)$;
    $z_k = \text{lookup}(Z, r_k)$;
    $z_k = \text{interpolate}(z_k, h_K, w_K)$;
    $f = f - \phi_k(z_k)$;
Return: multi-scale tokens R;

Algorithm 2: Multi-scale VQVAE Reconstruction

Inputs: multi-scale token maps R;
Hyperparameters: steps K, resolutions $(h_k, w_k)_{k=1}^{K}$;
$\hat{f} = 0$;
for $k = 1, \cdots, K$ **do**
    $r_k = \text{queue\_pop}(R)$;
    $z_k = \text{lookup}(Z, r_k)$;
    $z_k = \text{interpolate}(z_k, h_K, w_K)$;
    $\hat{f} = \hat{f} + \phi_k(z_k)$;
$\hat{im} = \mathcal{D}(\hat{f})$;
Return: reconstructed image $\hat{im}$;

METHOD: Visual Autoregressive Modeling (VAR)

Next-scale Prediction



METHOD: Implementation Details

- **VAR tokenizer:** VQVAE architecture + K extra convolutions (0.03M)

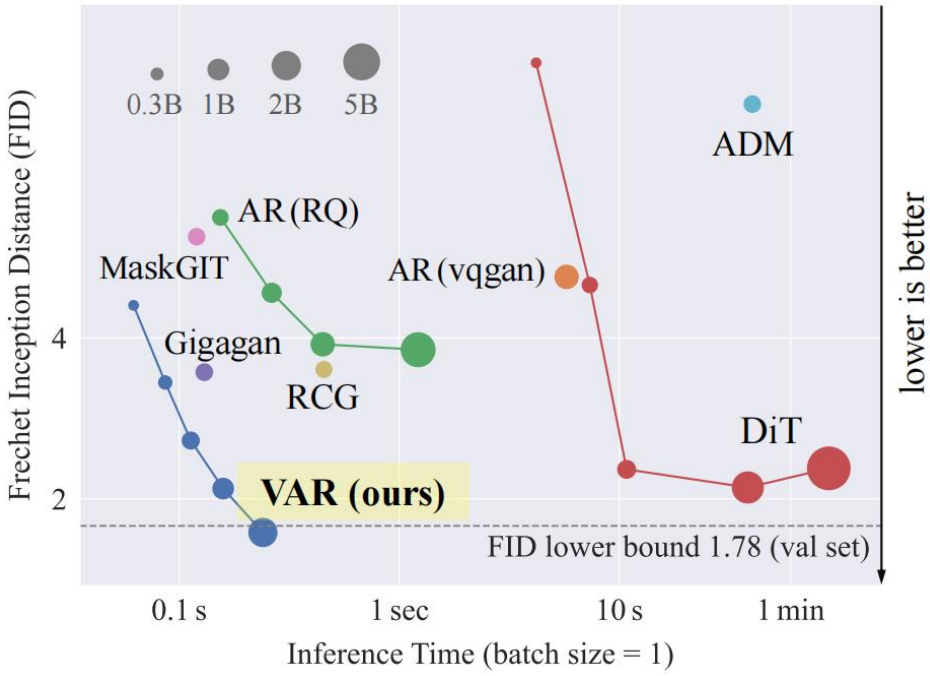
- **VAR transformer**
  - Standard decoder-only transformers akin to GPT-2
  - Adaptive Layer Normalization (AdaLN)
  - Normalizing queries and keys

OUTLINE

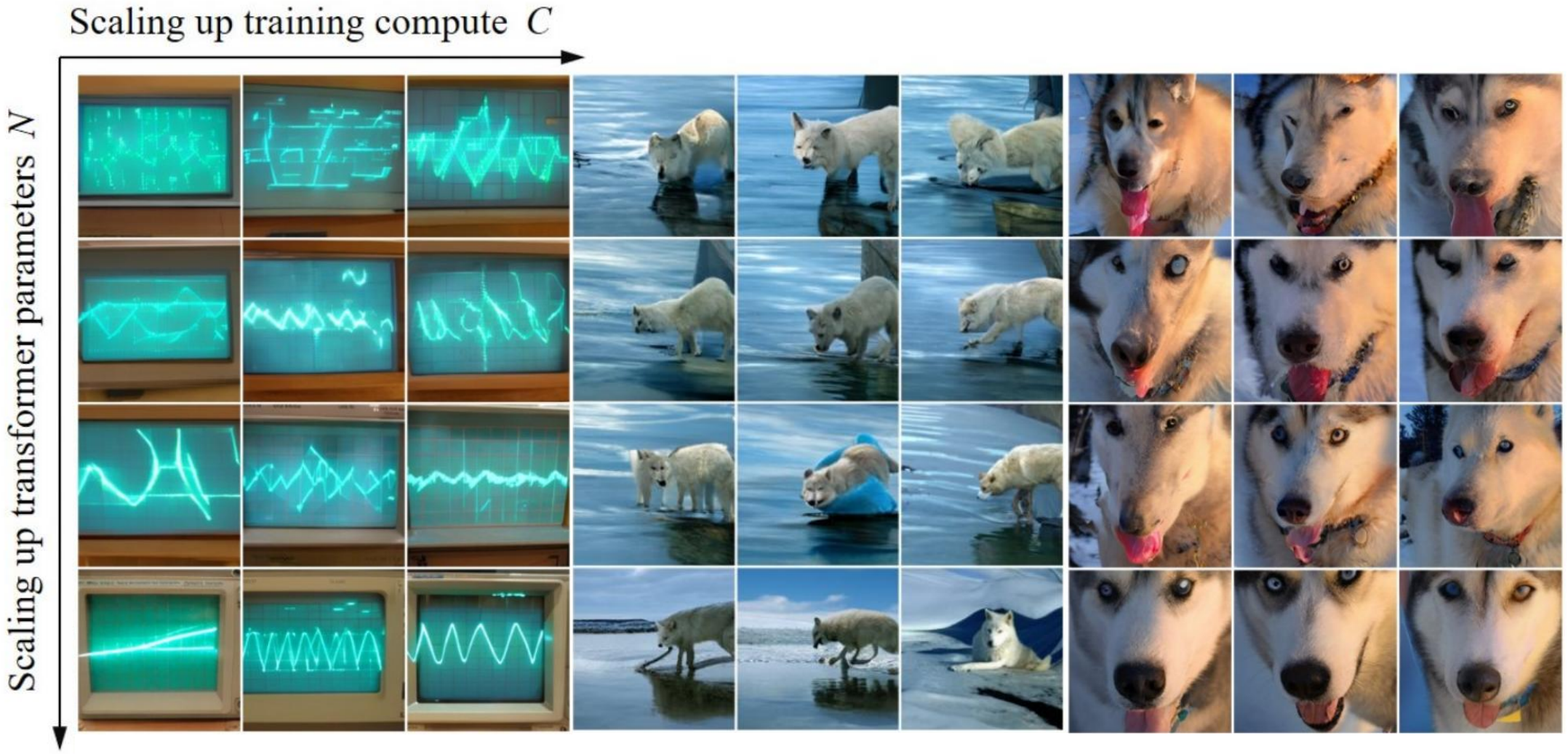
- Authorship
- Background
- Method
- Experiments
- Conclusion

EXPERIMENTS: Quanlitative

Type	Model	FID↓	IS↑	Pre↑	Rec↑	#Para	#Step	Time
GAN	BigGAN [7]	6.95	224.5	**0.89**	0.38	112M	1	–
GAN	GigaGAN [29]	3.45	225.5	0.84	**0.61**	569M	1	–
GAN	StyleGan-XL [57]	2.30	265.1	0.78	0.53	166M	1	0.3 [57]
Diff.	ADM [16]	10.94	101.0	0.69	0.63	554M	250	168 [57]
Diff.	CDM [25]	4.88	158.7	–	–	–	8100	–
Diff.	LDM-4-G [53]	3.60	247.7	–	–	400M	250	–
Diff.	DiT-L/2 [46]	5.02	167.2	0.75	0.57	458M	250	31
Diff.	DiT-XL/2 [46]	2.27	278.2	0.83	0.57	675M	250	45
Diff.	L-DiT-3B [2]	2.10	304.4	0.82	0.60	3.0B	250	>45
Diff.	L-DiT-7B [2]	2.28	316.2	0.83	0.58	7.0B	250	>45
Mask.	MaskGIT [11]	6.18	182.1	0.80	0.51	227M	8	0.5 [11]
Mask.	MaskGIT-re [11]	4.02	355.6	–	–	227M	8	0.5 [11]
Mask.	RCG (cond.) [38]	3.49	215.5	–	–	502M	20	1.9 [38]
AR	VQVAE-2† [52]	31.11	∼45	0.36	0.57	13.5B	5120	–
AR	VQGAN† [19]	18.65	80.4	0.78	0.26	227M	256	19 [11]
AR	VQGAN [19]	15.78	74.3	–	–	1.4B	256	24
AR	VQGAN-re [19]	5.20	280.3	–	–	1.4B	256	24
AR	ViTVQ [71]	4.17	175.1	–	–	1.7B	1024	>24
AR	ViTVQ-re [71]	3.04	227.4	–	–	1.7B	1024	>24
AR	RQTran. [37]	7.55	134.0	–	–	3.8B	68	21
AR	RQTran.-re [37]	3.80	323.7	–	–	3.8B	68	21
VAR	VAR-*d*16	3.60	257.5	0.85	0.48	310M	10	0.4
VAR	VAR-*d*20	2.95	306.1	0.84	0.53	600M	10	0.5
VAR	VAR-*d*24	2.33	320.1	0.82	0.57	1.0B	10	0.6
VAR	VAR-*d*30	1.97	334.7	0.81	**0.61**	2.0B	10	1
VAR	VAR-*d*30-re	**1.80**	**356.4**	0.83	0.57	2.0B	10	1
	(validation data)	1.78	236.9	0.75	0.67			

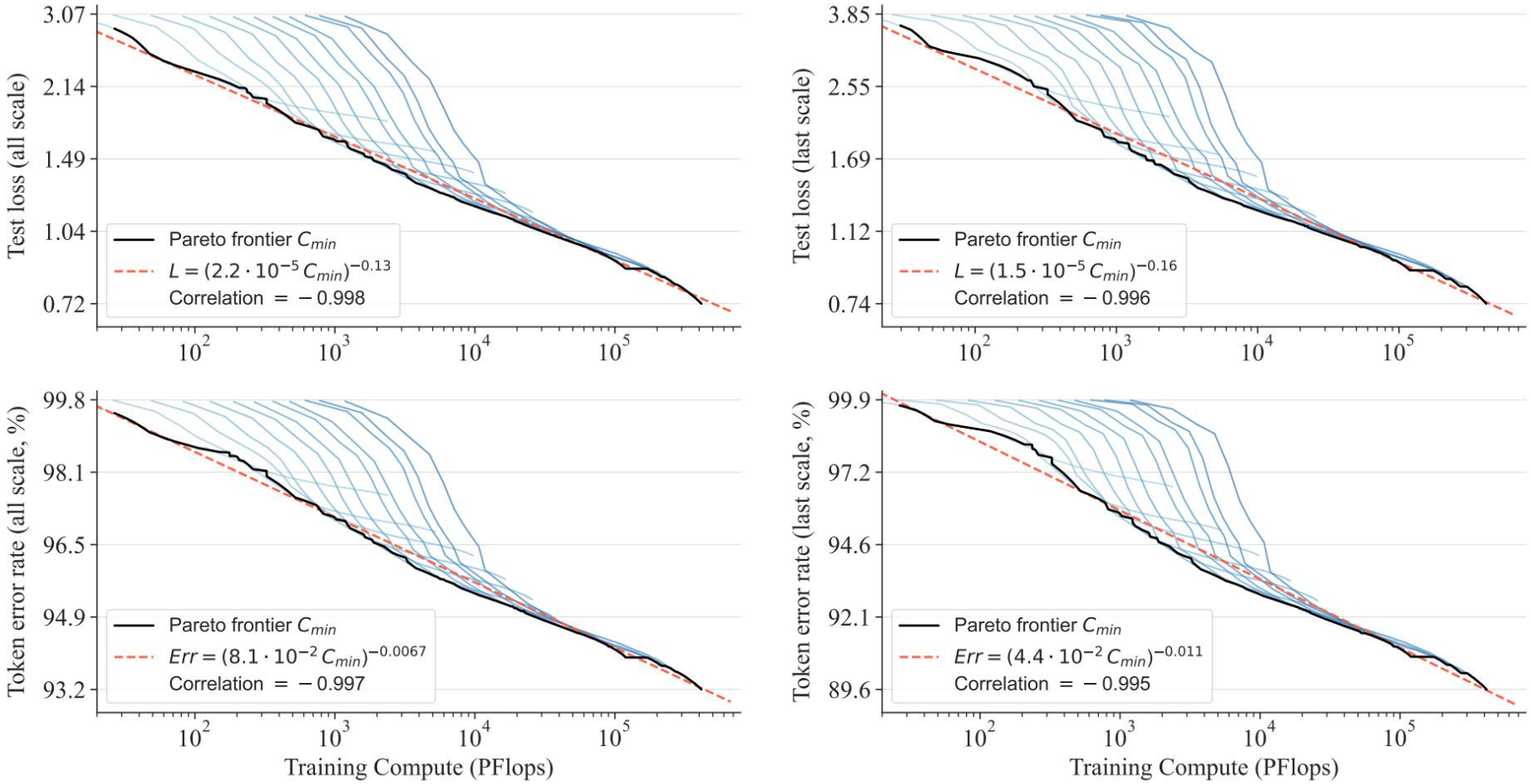


EXPERIMENTS: Visualization



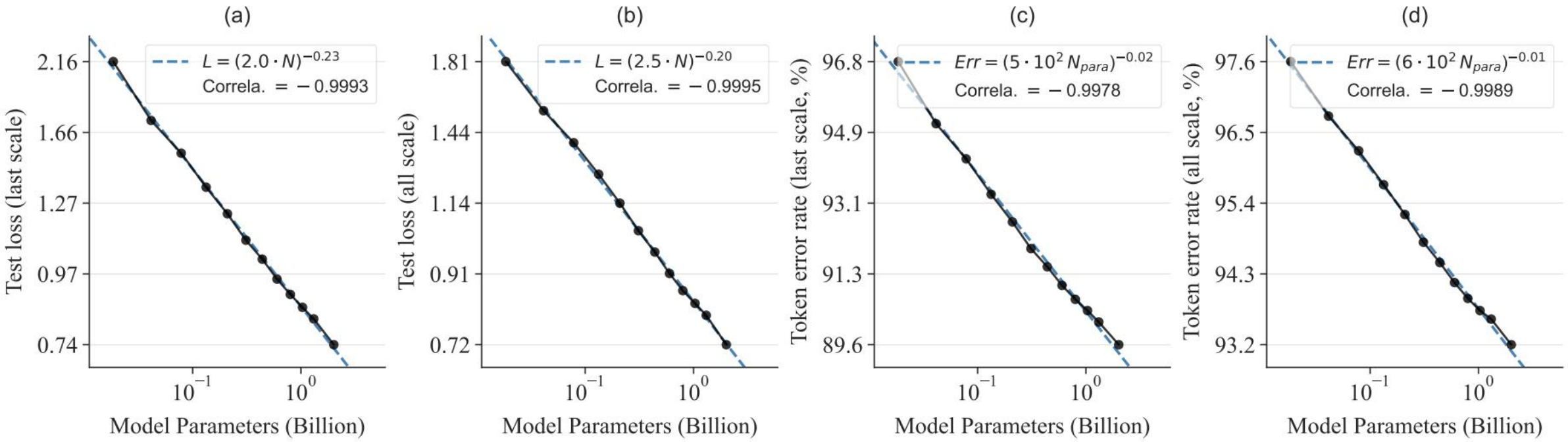
EXPERIMENTS: Scaling Laws

With Optimal Training Compute



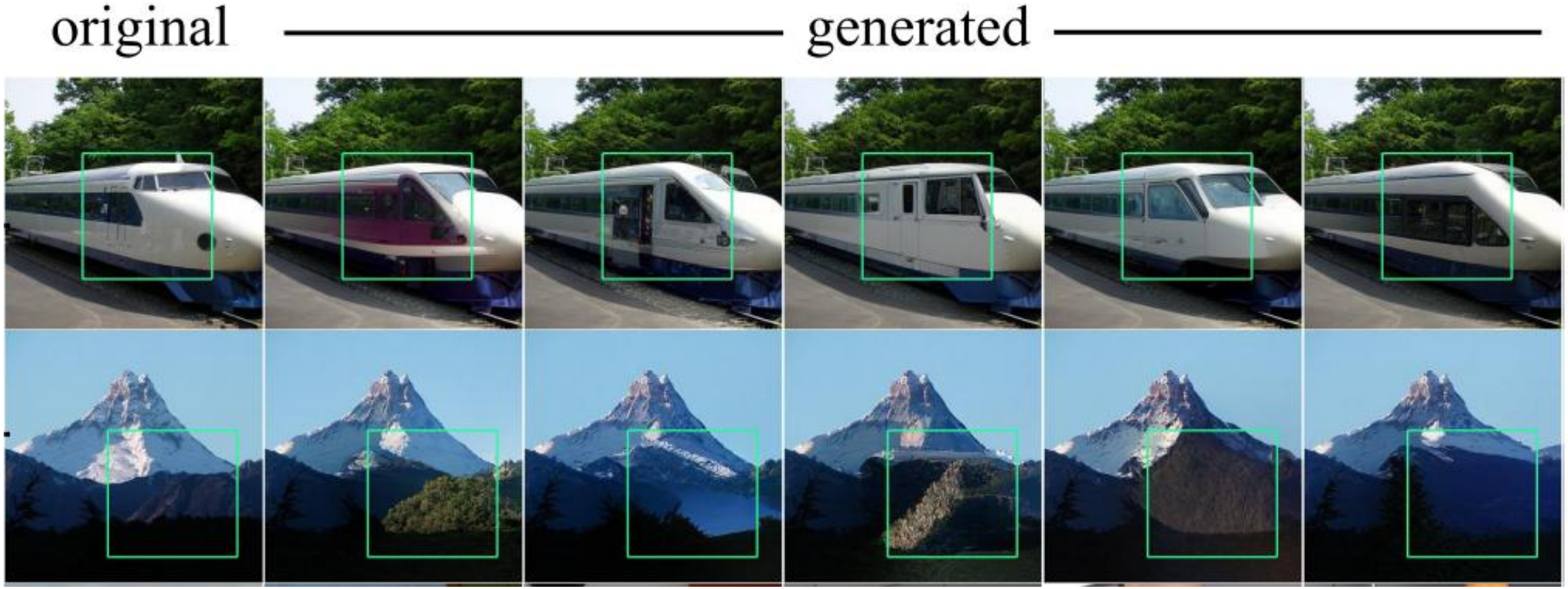
EXPERIMENTS: Scaling Laws

With Model Parameters



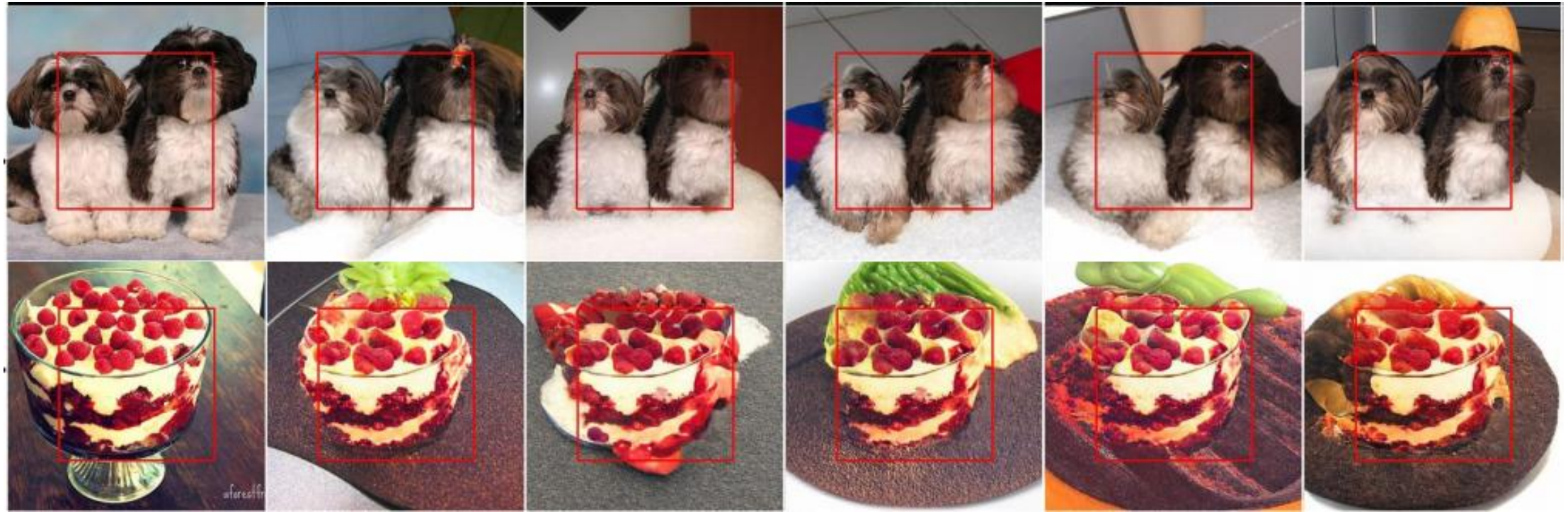
EXPERIMENTS: Zero-Shot Task

In-Painting



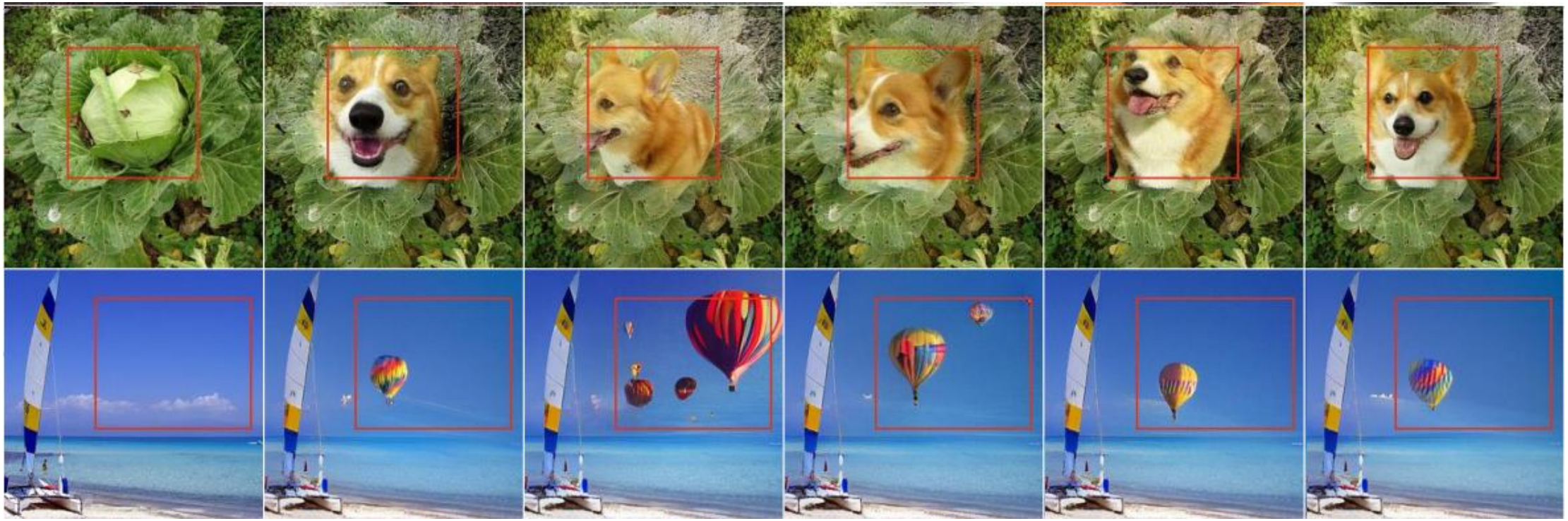
EXPERIMENTS: Zero-Shot Task

Out-Painting



EXPERIMENTS: Zero-Shot Task

Class-Cond Editing



EXPERIMENTS: Ablation Studies

	Description	Para.	Model	AdaLN	Top-k	CFG	Cost	FID↓	Δ
1	AR [30]	227M	AR	✗	✗	✗	1	18.65	0.00
2	AR to VAR	207M	VAR-$d16$	✗	✗	✗	0.013	5.22	−13.43
3	+AdaLN	310M	VAR-$d16$	✓	✗	✗	0.016	4.95	−13.70
4	+Top-k	310M	VAR-$d16$	✓	600	✗	0.016	4.64	−14.01
5	+CFG	310M	VAR-$d16$	✓	600	2.0	0.022	3.60	−15.05
5	+Attn. Norm.	310M	VAR-$d16$	✓	600	2.0	0.022	3.30	−15.35
6	+Scale up	2.0B	VAR-$d30$	✓	600	2.0	0.052	1.73	−16.85

OUTLINE

- Authorship
- Background
- Method
- Experiments
- Conclusion

CONCLUSION

- Introduce a new visual generative framework named Visual AutoRegressive modeling
- Make language-model-based AR models first surpass strong diffusion models
- Observe a clear power-law relationship

CONCLUSION: Future Work

- Advancing tokenizer structure
- Text-prompt generation
- Video generation

Thanks for listening!