Bridging Continuous and Discrete Tokens for Autoregressive Visual Generation

Supplementary Material

Appendix

The supplementary material includes the following additional information:

- Sec. A provides implementation details for TokenBridge.
- Sec. B presents speed comparison of our token prediction against diffusion-based head.
- Sec. C evaluates generalization to different VAE and AE architectures.
- Sec. D discusses limitations and broader impacts.
- Sec. E showcases additional image generation results.

A. Implementation Details for TokenBridge

We train our models on the ImageNet-1K [7] training set, consisting of 1,281,167 images across 1,000 object classes. We adopt the VAE tokenizer from [18] and apply our dimension-wise quantization with B=64 levels to its continuous features. For the autoregressive model architecture, we follow MAR [18], with our L model consisting of 32 transformer blocks (width 1024) and H model using 40 blocks (width 1280). Our dimension-wise autoregressive head uses 1024 hidden dimensions with 4 layers for the L model and 6 layers for the H model. At inference time, we employ temperature sampling and classifier-free guidance [13] to enhance generation quality. The detailed training and sampling hyper-parameters are listed in Tab. 4.

config	value
training hyper-params	
optimizer	AdamW [19]
learning rate	4e-4
weight decay	0.02
optimizer momentum	(0.9, 0.95)
batch size	2048
learning rate schedule	cosine decay
warmup epochs	200
ending learning rate	0
total epochs	800
dropout rate	0.1
attn dropout rate	0.1
class label dropout rate	0.1
precision	bfloat16
EMA momentum	0.9999
max grad norm	1.0
sampling hyper-params	
temperature	0.97(L) / 0.91(H)
CFG class dropout rate	0.1
guidance scale	3.1 (L) / 3.45 (H)

Table 4. **Detailed hyper-parameters for TokenBridge.**

B. Speed Comparison of Token prediction

We compare the speed of our dimension-wise prediction approach with MAR's [18] diffusion-based approach. Table 5 shows the results.

As shown in Table 5, our approach is 5.94× faster than MAR's [18] diffusion-based [35] token prediction. This efficiency advantage comes from our dimension-wise autoregressive prediction strategy that directly generates discrete tokens without iterative sampling procedures. Although our method requires sequential prediction steps (one per channel), the lightweight design of our AR head and the ability to utilize KV cache in transformers maintain high efficiency compared to diffusion sampling.

The number of prediction steps in TokenBridge corresponds to the VAE [15] channel count (16 in our implementation). With newer architectures like SDXL's [25] VAE that use only 4 channels, our approach would require even fewer steps.

Method	Time (ms)
Diffusion (MAR)	311.25 ± 1.85
AR (Ours)	52.42 ± 0.57

Table 5. **Comparison of single image token prediction time.** All measurements conducted with batch size 1 on an NVIDIA A100 GPU, averaged over 100 runs. Our method achieves a 5.94× speedup over MAR's diffusion sampling (100 steps).

C. Generalization to Different VAE and AE Architectures

To evaluate the generalization of our post-training quantization approach, we select two representative alternative autoencoders for evaluation: VAVAE [47], a state-of-the-art VAE with representation alignment, and DCAE [4], achieving high compression rates without KL loss constraints. Fig. 9 visualizes the latent feature distributions of different autoencoders. Although the value ranges differ across architectures, all exhibit similar near-Gaussian distributions. This consistency validates that the bounded, approximately Gaussian property, independent of specific architectural designs or training constraints like KL regularization. As described in the Method section, even linear quan-

tization achieves good reconstruction results, demonstrating the robustness of our approach across different quantization schemes.

Tab. 6 shows reconstruction results after applying our quantization with corresponding rescaling and quantization granularity. Our method successfully preserves reconstruction quality across different architectures: VAVAE achieves identical performance (rFID=0.28) to its continuous baseline using B=128 and r=3.5, while DCAE matches its baseline (rFID=0.77) with B=64 and r=8. These results demonstrate that our post-training quantization approach generalizes effectively across diverse autoencoder architectures while maintaining reconstruction fidelity.

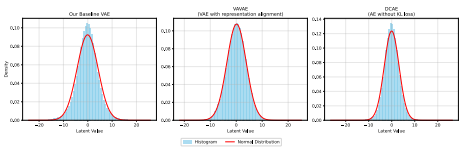

Figure 9. **Latent distributions of different autoencoders.** Despite architectural differences and training objectives, all three models exhibit similar near-Gaussian distributions, validating the generalizability of our quantization approach.

AE	Ori. FID	B	Range	TokenBridge FID
VAVAE	0.28	128	[-3.5, 3.5]	0.28
DCAE	0.77	64	[-8, 8]	0.77

Table 6. **Reconstruction quality across different autoencoder architectures.** Our post-training quantization preserves reconstruction fidelity when applied with appropriate parameters.

D. Limitations and Broader Impacts

Limitations. Our approach inherits limitations from the underlying VAE [15] model. The representation quality of the pretrained VAE directly affects our reconstruction fidelity and generation capabilities. We note that further improvements in continuous tokenizer would directly benefit our approach.

Broader Impacts. Our work demonstrates that standard autoregressive modeling with cross-entropy loss can achieve quality comparable to more complex approaches. This finding may encourage simpler model designs in visual generation tasks and facilitate unified multimodal modeling based on autoregressive frameworks. Like all generative models, TokenBridge may reflect biases present in training data and could potentially be misused to create misleading content, which warrants careful consideration in deployment.

E. More Visualization Results

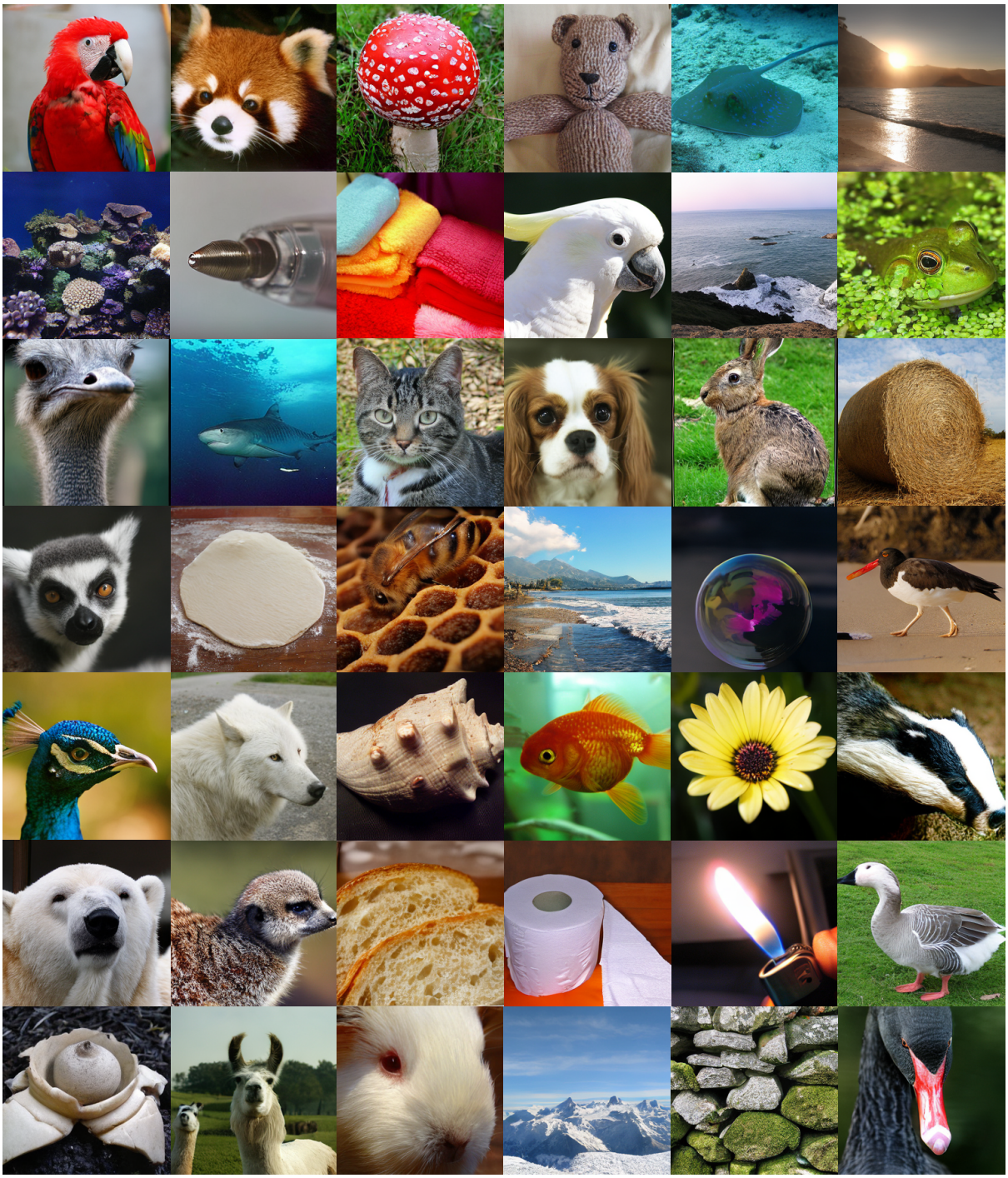

Figure 10. **Additional image generation results of TokenBridge across different ImageNet [7] categories.**