RADAR: VQ-VAE decoder of VAR is a good student for Restoring Against Degradation by Acceleration

Supplementary Material

6. Superior Quality–Throughput Trade-Off by RADAR

Table 5. Comparison of existing acceleration methods for VAR on quality and throughput benchmarks.

Methods	#Params	Throughput↑	FID↓	IS↑	Precision↑	Recall↑
VAR(d=24)	1.0B	32.21it/s	2.11	302.5	0.82	0.58
+CoDe(N=8)	1.0B+0.3B	43.78it/s	2.53	260.5	0.81	0.56
+MVAR	1.0B	37.16it/s	2.20	297.3	0.83	0.56
+FastVAR	1.0B	39.60it/s	2.44	281.4	0.79	0.57
+RADAR(ours)	1.0B	57.93it/s	2.19	298.2	0.83	0.56
VAR(d=30)	2.0B	22.75it/s	1.95	310.2	0.83	0.59
+CoDe(N=8)	2.0B+0.3B	29.04it/s	2.27	294.0	0.82	0.57
+MVAR	2.0B	25.29it/s	2.14	301.7	0.81	0.56
+FastVAR	2.0B	32.47it/s	2.21	304.6	0.81	0.55
+RADAR(ours)	2.0B	35.65it/s	2.01	306.2	0.85	0.58

In the additional experiment shown in Tab. 5, we apply several recent acceleration methods to the VAR model [29] and conduct 256×256 ImageNet-1K [7] image generation. For throughput measurements, the batch size is set to 32, whereas for image-quality evaluation, the batch size is fixed to 10 to ensure divisibility when generating 50 images per class. For CoDe [4], we follow its default setting without retraining, set N, the number of drafter steps, to 8. For MVAR [36], we perform an additional 80-epoch fine-tuning of full VAR Transformer.

As shown in Tab. 5, our method achieves the most competitive results in terms of both throughput and generation performance. Although CoDe and FastVAR [10] also obtain noticeable speedups, they come with non-trivial limitations. CoDe requires maintaining an additional small-scale VAR Transformer in memory and struggles to further accelerate models at smaller size. FastVAR exhibits substantial performance degradation under accelerated inference. MVAR not only incurs an expensive full-model retraining cost but also yields only marginal throughput gains. In contrast, our method explicitly addresses the performance degradation induced by aggressive acceleration and effectively enhances the model's robustness against such degradation. All our designs allow RADAR to adopt more aggressive acceleration configurations during inference while achieving a more decent throughput–quality trade-off.

7. Minimal Semantic Loss by SCA-Mask

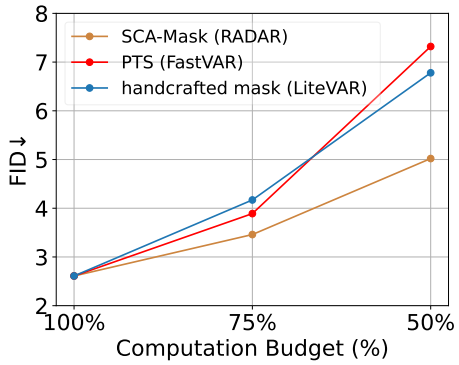

Figure 9. The generation quality comparison of different mask strategy under given computation budget.

As shown in Fig. 9, we compare three mask strategies under a fixed computational budget: the SCA-Mask used in our RADAR, the Pivotal Token Selection (PTS) in FastVAR based on deviation from the mean [10], and the handcrafted mask in LiteVAR [34]. VAR-d20 is used as the base inference model, and FID is used as the performance metric. When the available computational budget decreases, all mask strategies exhibit a degradation in generation performance. Nevertheless, our SCA-Mask consistently achieves the best performance in all budget settings, due to its ability to effectively measure the semantic importance of different local regions. The PTS performs slightly better than the handcrafted mask when the budget is reduced to 75%, but becomes worse when the budget is reduced to 50%. This may due to the fact that the value of deviation from the mean is largely influenced by the token's own activation values, which do not necessarily reflect its global importance. In contrast, our SCA-Mask strategy provides a more reliable preservation of generation quality by comprehensively assessing the effective interactions between each tile.

8. Highly Efficient Implementation for PAA

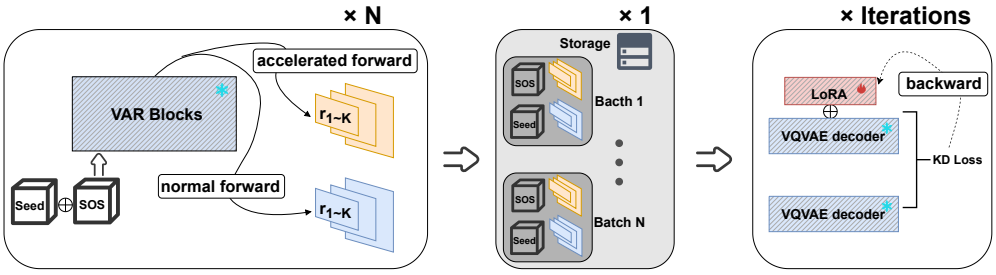

Figure 10. PAA pipeline implemented with latent representation cache to maximally decouple transformer-side forward and decoder backward. N denotes the number of batches.

Inspired by TinyViT [32], we further simplify the two-branch pipeline of Post-Acceleration Adaption (PAA) to minimize its overall computational cost. Specifically, since only a subset of parameters is learnable in PAA, the gradients from the distillation loss do not need to back-propagate through the VAR transformer. This allows all intermediate activations on the transformer-side to be discarded without any cost. As illustrated in Fig. 10, we fully decouple the VAR transformer and the VQ-VAE within PAA. We first use the VAR transformer to pre-generate all required high-quality and low-quality latent representations in batch, and store each latent sequence together with its corresponding class label, prompt, and random seed on disk. These high/low-quality latent representations are then fed into the frozen VQ-VAE decoder or the frozen VQ-VAE decoder equipped with LoRA [12], respectively, where gradients are used solely to update the LoRA parameters. Under this design, the peak memory footprint of the entire PAA pipeline becomes exactly identical to that of standard VAR inference, yielding up to 50% improvement in adaption throughput. In addition, thanks to transformer being completely frozen in our method, cached latent sequences can be reused across multiple training iterations for best efficiency.

9. Analysis of Failure Cases and Limitations of PAA

As illustrated in Fig. 11, the degradations induced by the aggressive mask can be effectively mitigated by PAA in most cases. These degradations primarily include distributional shifts and mild structural collapse in the latent space. Furthermore, it is important to clarify that under excessively aggressive acceleration settings, the latent representations may experience severe collapse. In such scenarios, the degradation typically manifests itself as a global structural collapse of the generated image, rather than a transformation into semantically meaningful images belonging to other categories. Due to the complete loss of coherent structural information, PAA cannot recover the original content in these extreme cases. We will explore explicitly incorporating labels or prompts into the input of PAA to enable a more comprehensive restoration in future work.

10. More Qualitative Results

In Fig. 12 and Fig. 13, we present additional qualitative results of the original VAR and the degraded VAR equipped with RADAR at 512*512 and 256*256 resolutions, respectively. For the 256*256 resolution setting, eight images are generated per class, while for the 512*512 resolution setting, four images are generated per class. All random seeds are kept identical between VAR and RADAR to ensure a fair visual comparison. As shown by these results, our method preserves the global structure and maintains fine-grained details effectively, producing outputs that differ only marginally from the original VAR.

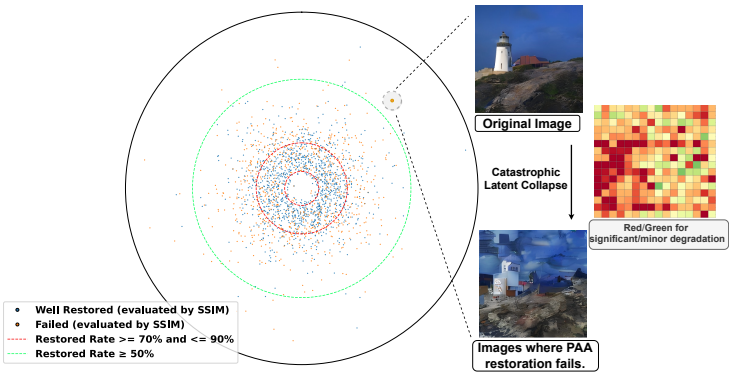

Figure 11. **Left:** Scatter plot visualization of 2000 restoration instances in polar coordinates. The radial distance from a scattered points to the original point is proportional to its$||\hat{f_n} - \hat{f_{ac}}||$. **Right:** Visualization of a failed case, including a heatmap visualization for $\|\hat{f_n} - \hat{f_{ac}}\|$ of 2D token grid. Best viewed in color with zoomed in.

Figure 12. Qualitative comparison at 512*512 resolution between the original VAR and degraded VAR w/ RADAR. Each class is visualized with two rows: the first row shows outputs of the original VAR, while the second row shows the degraded VAR restored by RADAR.

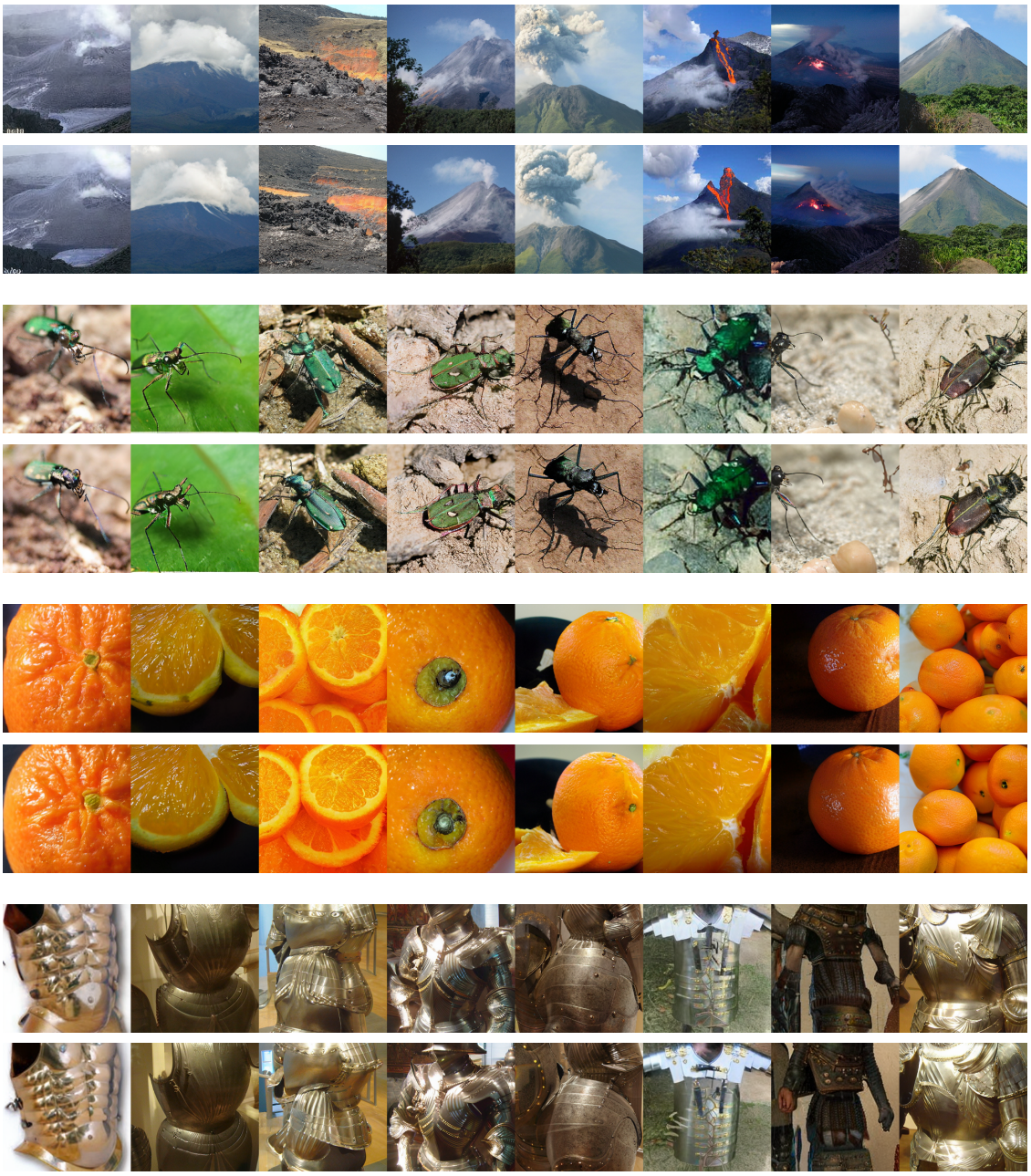

Figure 13. Qualitative comparison at 256*256 resolution between the original VAR and degraded VAR w/ RADAR. Each class is visualized with two rows: the first row shows outputs of the original VAR, while the second row shows the degraded VAR restored by RADAR.