

---

# Supplementary for Towards Better & Faster Autoregressive Image Generation: From the Perspective of Entropy

---

Anonymous Author(s)

Affiliation

Address

email

## A Introduction

We first provide additional details of our method, including parameter settings and further descriptions, in Sec. B. Then, we present extended experimental results in Sec. C. In Sec. D, we conduct deeper analyses on entropy in relation to model behavior and image content, along with more visualizations of entropy maps. Sec. E discusses potential limitations and future directions. Lastly, we include more visual comparisons of the proposed method in Sec. F.

## B Additional details of method

### B.1 Hyperparameter settings in Sec. 3.2 in maintext

In Sec. 3.2 in maintext, we propose to dynamically control the sampling temperature based on entropy. However, due to significant differences between base models, it is difficult to apply the same parameters across all settings. Therefore, we list the detailed parameters for each model in Table 1. For undertrained models such as LlamaGen stage1, higher randomness is required at low-entropy stages to avoid generating large areas of repetitive tokens. In contrast, well-trained models benefit from a smoother temperature schedule.

Table 1: Hyperparameter settings of different models.

	$T_0$	$\alpha$	$\theta$
LlamaGen	2.5	3.0	0.6
Lumina-mGPT	2.0	2.5	0.6
Meissonic	2.5	3.0	0.7
STAR	2.5	3.0	0.5

### B.2 Detailed description of speculative decoding in images

We accelerate inference based on existing speculative decoding schemes [1] in Sec. 3.4 in maintext, thereby further reducing inference cost without sacrificing output quality. Due to space constraints, we did not elaborate on the baseline speculative decoding methods in the main text. Here, we provide more details.

This method aims to accelerate auto-regressive text-to-image generation by allowing multiple tokens to be generated in parallel without training. Inspired by speculative decoding, SJD introduces a probabilistic acceptance criterion that compares the confidence of draft tokens from two consecutive iterations. In each iteration  $j$ , given a draft token  $x_i^{(j)}$ , SJD computes its acceptance probability based

on the ratio between two conditional probabilities:

$$r < \min \left( 1, \frac{p_{\theta}(x_i^{(j)} | x_{1:i-1}^{(j)})}{p_{\theta}(x_i^{(j)} | x_{1:i-1}^{(j-1)})} \right), \quad (1)$$

where  $r \sim \mathcal{U}[0, 1]$ . Accepted tokens are fixed, while the others are resampled from a calibrated distribution:

$$x_i^{(j+1)} \sim \frac{\max(0, p_{\theta}(x | x_{1:i-1}^{(j)}) - p_{\theta}(x | x_{1:i-1}^{(j-1)}))}{\sum_x \max(0, \cdot)}. \quad (2)$$

This allows high-randomness sampling, crucial for image diversity, while significantly reducing decoding steps. SJD operates in a windowed, iterative manner and supports optional spatially-informed token initialization to further improve efficiency.

## C Additional experimental results

### C.1 Entropy & top- $p$ and temperature

In the main text, we analyze the relationship between our entropy-based sampling strategy and existing sampling parameters such as CFG and top- $K$ . By combining our method with these parameters, we observe improved robustness, reducing sensitivity to hyperparameter choices and yielding better FID and CLIP-Score. Here, we further examine other sampling parameters—top- $p$  and temperature—which are rarely used in autoregressive models due to their tendency to distort the output distribution and severely degrade either FID or CLIP-Score. Comparative results between our method and the baseline are shown in Fig. 1.

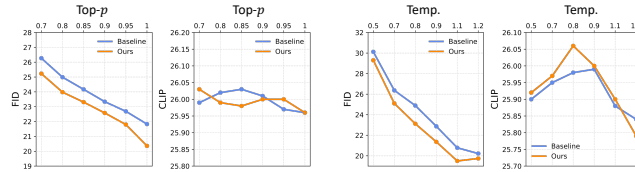


Figure 1: Combination of our sampling strategy with existing methods (Top- $p$ , temperature). “Temp.” is short for temperature.

### C.2 Additional comparison with top- $K$ and CFG

In Sec. 4.4 of the main paper, we discussed the differences between our method and existing sampling strategies (Top- $K$  and CFG) using LlamaGen. Here, we provide additional comparisons on other models to analyze the relationship between entropy-aware temperature and these conventional sampling approaches, results are presented in Fig. 2 and Fig. 3. Consistent with our observations in Sec. 4.4 in maintext, the proposed strategy mitigates performance fluctuations caused by hyperparameter choices (e.g., CFG and top- $K$ ), leading to a better balance between fidelity and text-image alignment.

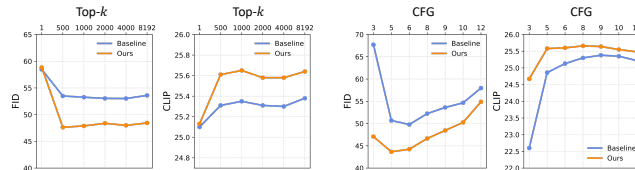


Figure 2: Combination of our sampling strategy with existing methods (Top- $k$ , CFG) on Meissson.



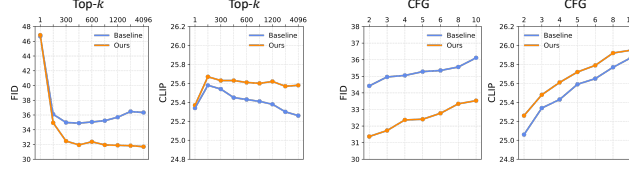


Figure 3: Combination of our sampling strategy with existing methods (Top- $k$ , CFG) on STAR.

## 46 D Additional discussion about entropy

### 47 D.1 Entropy & generative models

#### 48 D.1.1 Visualization of entropy & images

49 Due to space limitations in the main text, we did not provide extended entropy visualizations and  
 50 analysis. Here, we include additional entropy maps for LlamaGen and Lumina-mGPT. See Fig. 4 and  
 51 Fig. 5.

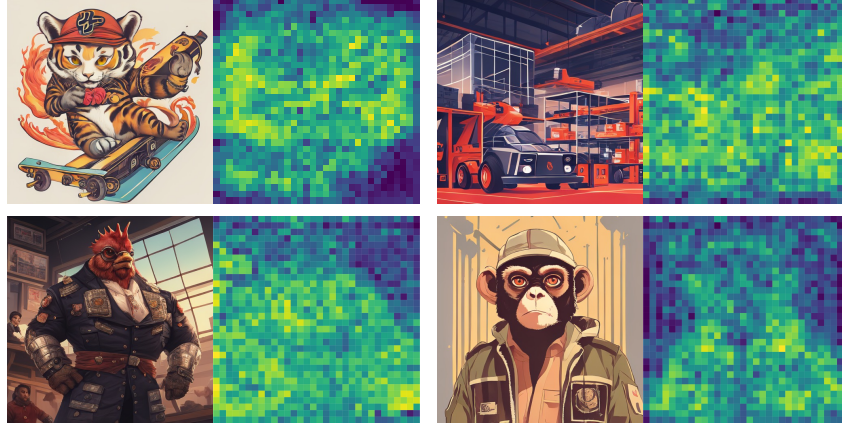


Figure 4: Entropy visualization of LlamaGen.

#### 52 D.1.2 Mask-prediction models

53 We provide additional entropy-based analysis of the mask model. Since the generation involves  
 54 multiple timesteps, where a subset of tokens is accepted at each step based on previously generated  
 55 content, we compute the entropy of accepted tokens at each timestep and aggregate them into a final  
 56 entropy map. As shown in Fig. 6, applying the proposed entropy-based temperature leads to a more  
 57 spatially balanced entropy distribution and enables richer image content while maintaining generation  
 58 stability.

59 In addition, we further analyze the average entropy of tokens accepted at each timestep, as shown in  
 60 Fig. 7. As discussed in the main text, due to the confidence-based token selection strategy, tokens  
 61 accepted in earlier steps tend to have lower entropy, since they are more likely to receive high  
 62 confidence scores. In contrast, tokens accepted in later steps ( $>60$ ) exhibit significantly higher entropy.  
 63 Moreover, more tokens are accepted in these later stages, which increases the risk of violating the  
 64 autoregressive assumption that spatially adjacent tokens should be sampled as independently as  
 65 possible. This may lead to degraded image quality. Therefore, adopting a more conservative sampling  
 66 strategy for these high-entropy tokens could help improve the overall generation quality.

#### 67 D.1.3 Scale-wise models

68 For the scale-wise model, the generation process constructs a complete image by predicting logits  
 69 maps at multiple scales. Each scale is conditioned on the residuals from the preceding scales, meaning  
 70 that the sum of the feature maps generated at all scales is passed through the detokenizer to form

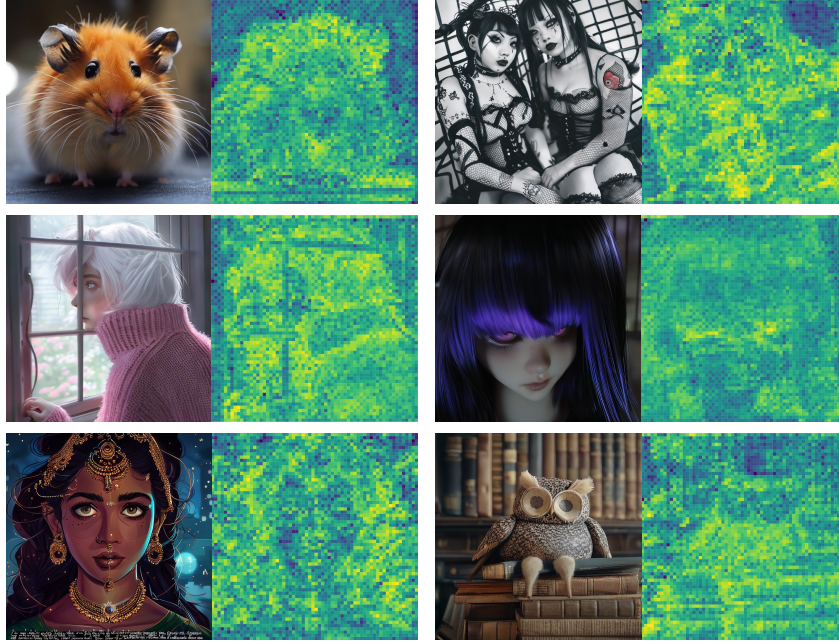


Figure 5: Entropy visualization of Lumina-mGPT.

the final output. In this generation paradigm, different scales exhibit distinct roles. Specifically, as described in [2], the earlier scales are responsible for generating the main structure of the image, while the later scales refine the result with fine details such as texture. We visualize the entropy maps of each scale during generation, as shown in Fig. 8. From scale 8 to scale 12, the model tends to focus more on the foreground, with significantly higher entropy observed in the regions corresponding to the primary subject. In contrast, at scale 13 and 14, there is no clear bias between foreground and background, indicating a more uniform attention across the image.

In addition, we compute the average entropy for each scale, as shown in Fig. 9. The later scales exhibit relatively higher entropy, while the earlier scales tend to have lower average entropy (however a decreasing trend is observed in the final two scales). This further indicates that different scales carry varying amounts of information.

## D.2 Entropy & generated contents

In practice, the logits are not simply positively correlated with the complexity of image content. We observe that regions with clear, well-defined content do not always exhibit high entropy; instead, their entropy typically falls within a moderate range (e.g., between 2 and 8). The more deterministic the content, the lower the entropy tends to be. In contrast, regions with entropy lower than 2 or higher than 8 often correspond to simple backgrounds or overly complex, unfaithful details. Especially for regions with entropy above 8, the generated details are frequently meaningless. This also explains why adjusting the logits in these low- and high-entropy areas, as discussed in our motivation experiment, does not significantly harm text-image alignment.

## D.3 Is entropy the best indicator for information?

From a theoretical perspective, the entropy of logits reflects the model’s confidence in predicting the current token. When the model is sufficiently trained—or when its capacity is strong—it may produce low entropy even in semantically important regions. In practice, we observe cases where foreground objects (e.g., faces) yield lower entropy than complex backgrounds (See Fig. 10). This suggests that the model’s confidence is not solely determined by information density, but also by the number of plausible token candidates in a region. For instance, highly structured areas like faces tend to have a unique correct token and thus low uncertainty, despite containing rich semantic information.



Figure 6: Entropy visualization of Meissonic. Our entropy-based temperature leads to a more spatially balanced entropy distribution and enables richer image content.

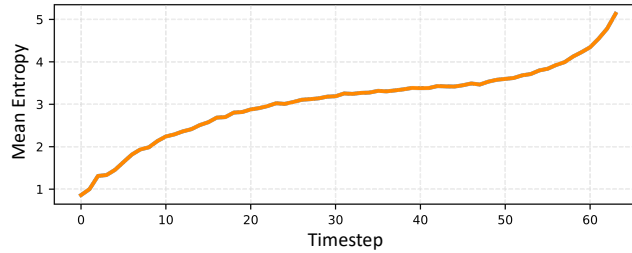


Figure 7: Mean entropy of each step from mask-prediction model. Values are averaged from  $\sim 100$  generated images.

99 In contrast, cluttered textures such as grass or foliage may allow for more varied token predictions,  
100 resulting in higher entropy.

101 Based on the above analysis, entropy may need to be combined with additional indicators to more  
102 accurately characterize the information distribution within an image. Specifically, more precise  
103 token-wise handling can be achieved by incorporating the similarity among top-ranked tokens in  
104 the logits distribution. For instance, if the entropy is low but the top tokens are not similar, the  
105 prediction can be deemed accurate; however, if the top tokens are highly similar under low entropy,  
106 the randomness at that position may need to be further increased. Conversely, under high-entropy  
107 conditions, a set of similar top tokens may indicate the existence of genuinely diverse possibilities.  
108 Moreover, analyzing the similarity of logits between adjacent tokens could help identify tokens that  
109 require more precise predictions—for example, if a token’s probability distribution significantly



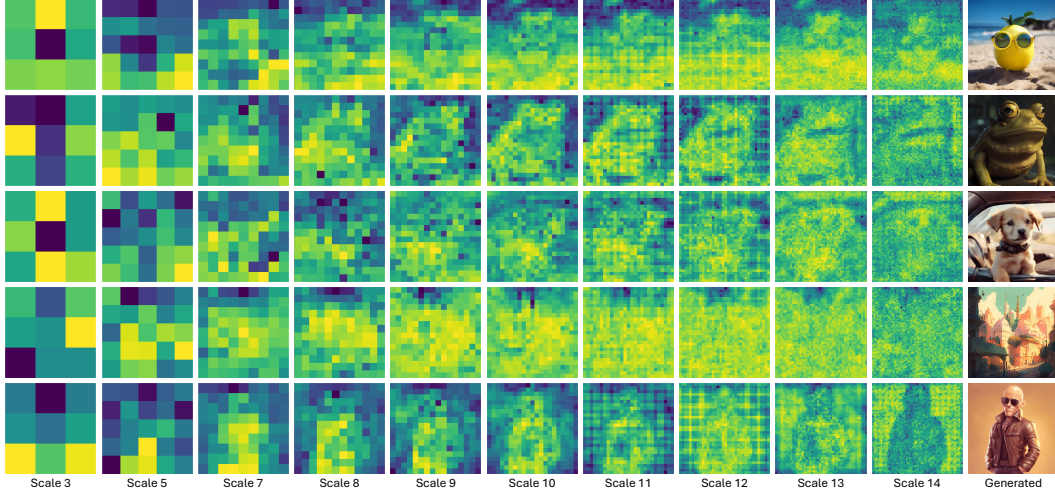


Figure 8: Entropy visualization of STAR. .

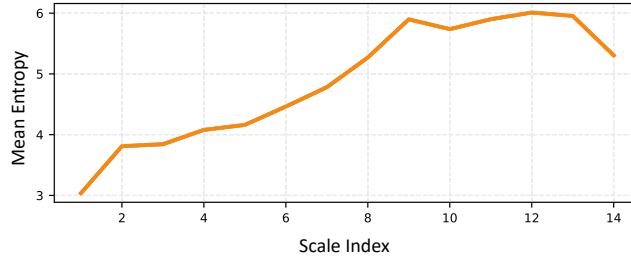


Figure 9: Mean entropy of each scale from scale-wise model. Values are averaged from  $\sim 100$  generated images.

differs from that of the previous token, it may warrant stricter sampling, regardless of its entropy level. We leave these directions for future exploration.

Moreover, since dynamic temperature only adjusts the randomness of the probability distribution (i.e., the variance of the logits) but not the location of its peak, further combining it with CFG may help achieve better performance.

## E Future works & limitations

### E.1 Broader impacts

This work is the first to explore the decoding problem in autoregressive visual generation, highlighting the inherent differences between image and text generation. While our approach may not be fully complete and still leaves room for improvement, we hope it can inspire future research to further investigate this issue and develop decoding strategies tailored specifically for visual generation, ultimately advancing unified multimodal generation.

### E.2 Future works

Currently, we propose a training-free sampling strategy for image generation by adaptively controlling sampling randomness based on the distribution of predicted logits. However, this approach is sensitive to hyperparameters, and due to significant differences across backbone architectures, optimal settings vary across models. Moreover, as a simple inference-time method built upon pretrained models, its performance gains may be limited for certain models.

In the future, this strategy could be integrated into the training framework for further performance improvement or acceleration. For example, it may be combined with early-exit mechanisms to

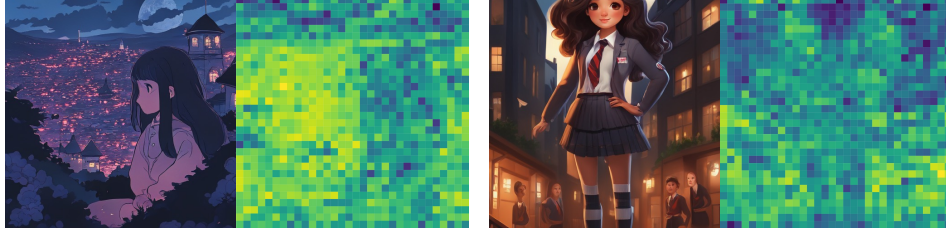


Figure 10: For some cases, the semantic-corresponding foreground contents may have smaller entropy.

130 allocate computation dynamically across tokens, or used to guide training by leveraging entropy to  
 131 focus more on informative regions, thus accelerating convergence.

### 132 E.3 Limitations

133 The proposed method mainly mitigates issues caused by inconsistent token sampling strategies  
 134 under varying information densities, but it does not enhance the intrinsic generation capability of  
 135 autoregressive models. The performance gain is model-dependent. If the base model is trained with  
 136 techniques that promote diverse token distributions, such as noise injection during training, or is  
 137 well-trained on large-scale datasets, the improvement tends to be limited. Moreover, for weak base  
 138 models, such as LlamaGen Stage 2, the method may offer little or no performance gain.

### 139 F Additional visual comparison

140 Due to space constraints in the main paper, we did not provide additional visualizations. Here, we  
 141 include further results illustrating the entropy-aware sampling behavior for LlamaGen, Lumina-mGPT,  
 142 Meissonic, and STAR, as well as acceleration visualizations for LlamaGen and Lumina-mGPT (see  
 143 Fig. 11–Fig. 15).



Figure 11: Visualization of LlamaGen.

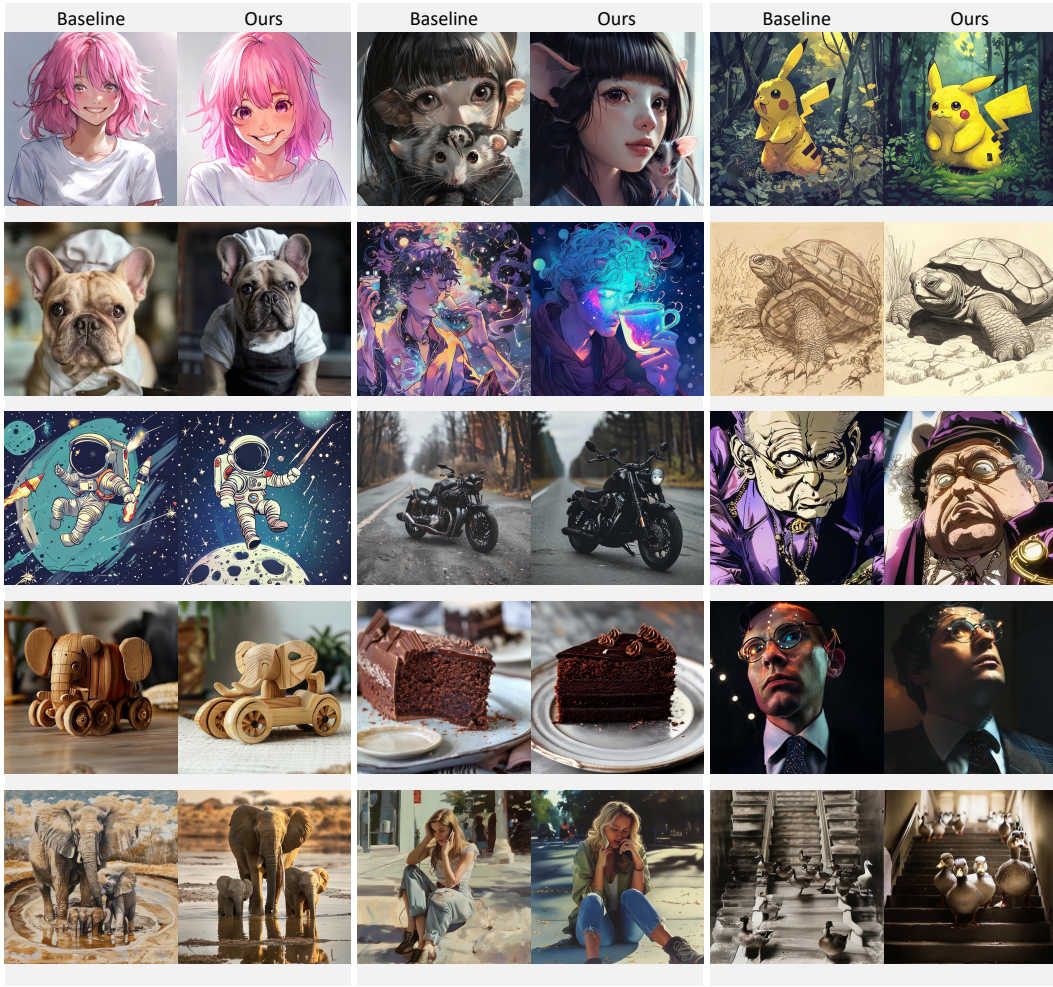


Figure 12: Visualization of Lumina-mGPT.



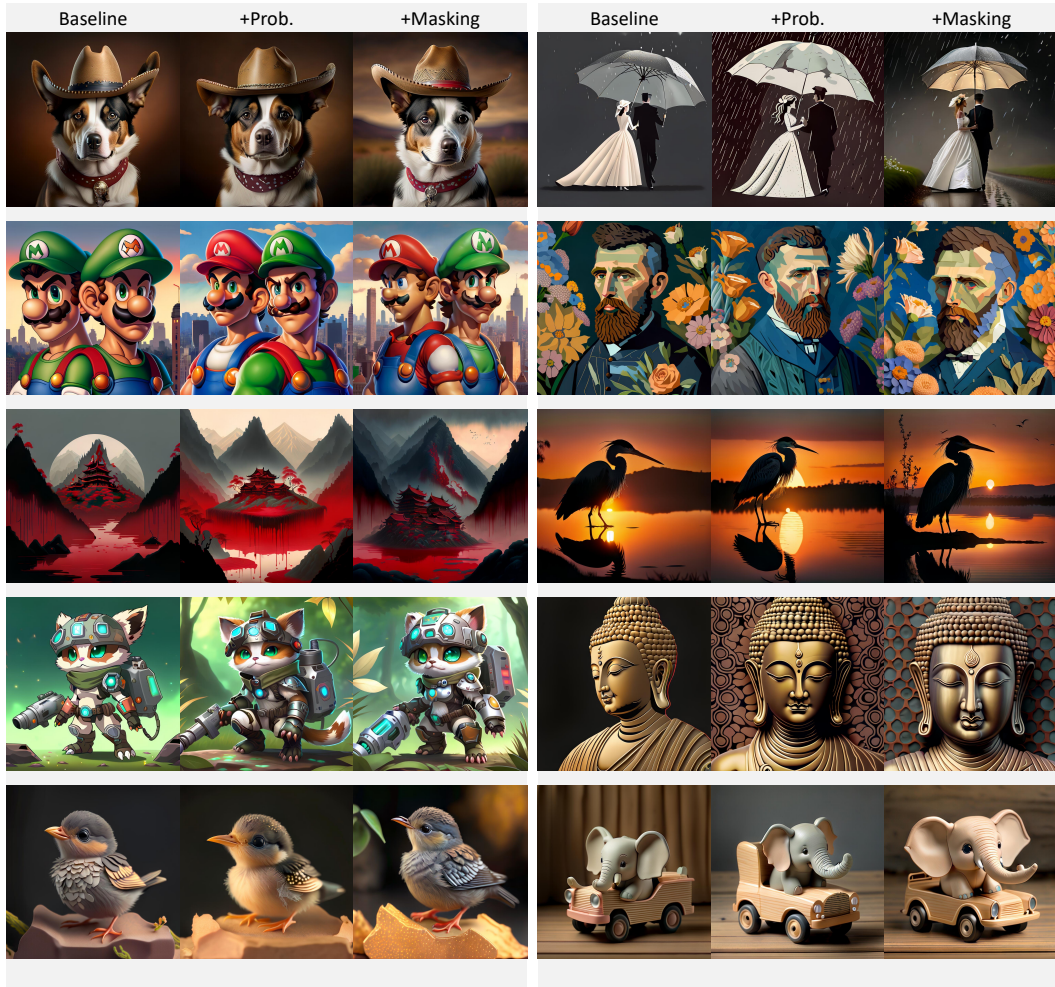


Figure 13: Visualization of Meissonic.

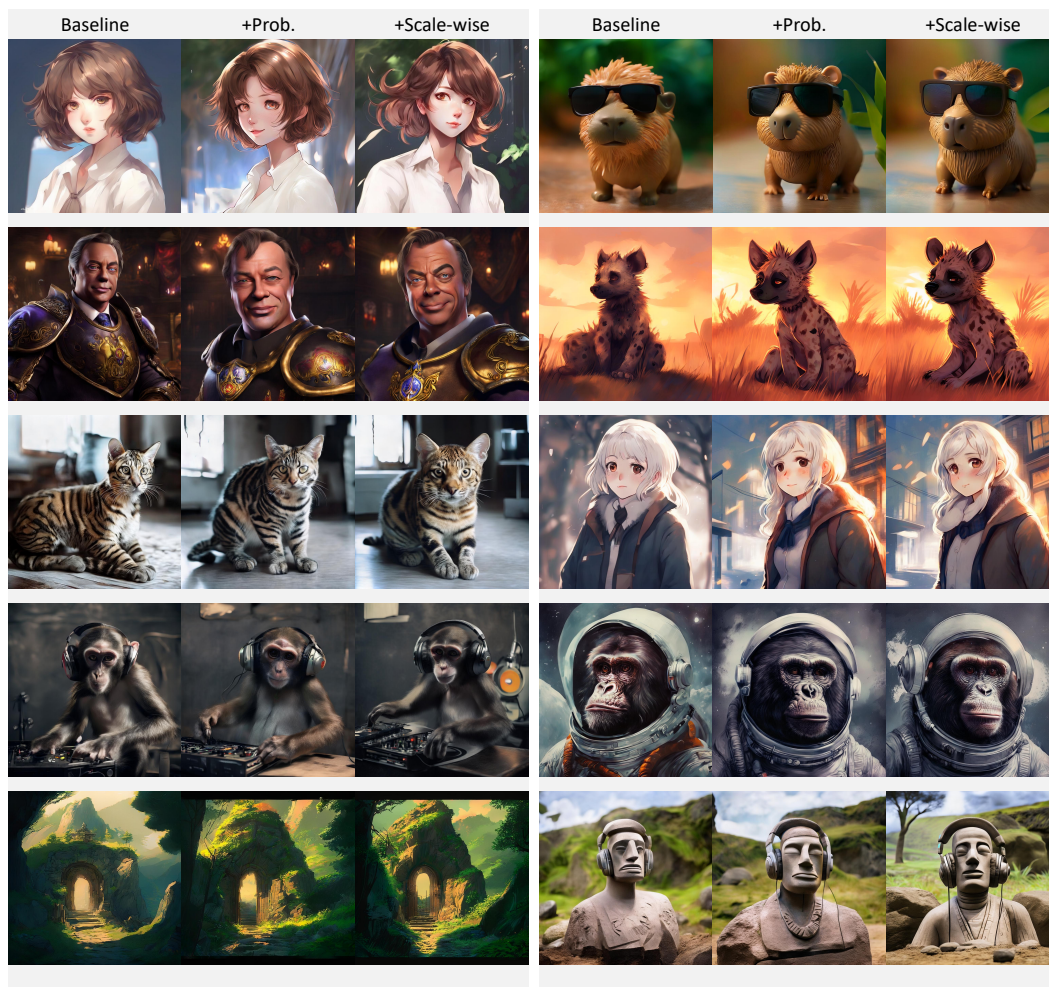


Figure 14: Visualization of STAR.



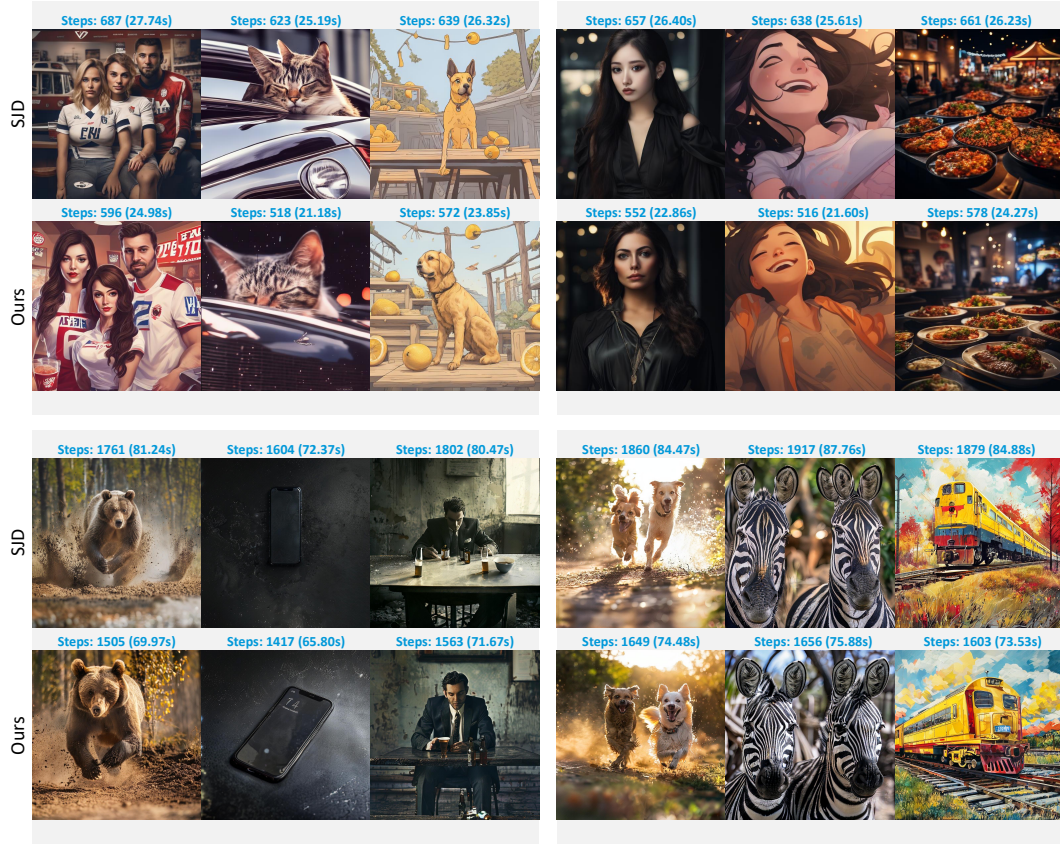


Figure 15: Visualization of AR acceleration.

## 144 **References**

- 145 [1] Yao Teng, Han Shi, Xian Liu, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Accelerating  
146 auto-regressive text-to-image generation with training-free speculative jacobi decoding. *arXiv preprint*  
147 *arXiv:2410.01699*, 2024.
- 148 [2] Yufei Wang, Lanqing Guo, Zhihao Li, Jiaxing Huang, Pichao Wang, Bihan Wen, and Jian Wang. Training-  
149 free text-guided image editing with visual autoregressive model, 2025.