Context-Aware Autoregressive Image Generation for Emerging Reasoning Properties

Jixuan Ying[1,2,*], **Haoyu Liu**[2,*,†], **Timing Yang**[1], **Tingyu Zhu**[2]
Predrag Neskovic[4], **Anand Bhattad**[1], **Cihang Xie**[3]
Zeyu Zheng[2,†], **Alan Yuille**[1], **Feng Wang**[1,†]
[1]Johns Hopkins University
[2]University of California, Berkeley [3]University of California, Santa Cruz
[4]Office of Naval Research
[*]Equal contribution [†]Corresponding authors
haoyuliu@berkeley.edu zyzheng@berkeley.edu wangf3014@gmail.com

Abstract

Autoregressive image generation has recently emerged as a competitive visual generation paradigm, yet its generation order is often predefined or heuristic, such as next-patch prediction, random-order prediction, or one-step generation for specific scales. We observe that these strategies can cause models to overlook important structural dependencies in images, thereby limiting both structural robustness and reasoning capability. In this work, we propose a context-aware autoregressive generation framework, where the model dynamically determines the next token location based on global context and current local information. Rather than following a fixed or random order, our method allows the generation process to adapt to image structure and semantic dependencies. Extensive experiments show that context-aware generation significantly improves structural robustness across different image AR paradigms, including masked autoregressive and scale-wise autoregressive models. Beyond quantitative improvements, we further observe emerging reasoning properties: the model can better understand complex semantic instructions and generate images that more faithfully follow the given guidance. These findings suggest that generation order is a critical yet underexplored factor in image autoregressive modeling, and that context-aware decoding can unlock stronger reasoning-aware visual generation capabilities. Code is available at https://github.com/Hpesojyjx/Saccades-AR.

1 Introduction

Inspired by the success of next-token prediction in natural language processing, recent years have witnessed a growing interest in autoregressive (AR) image generation. Early efforts directly transferred the language modeling recipe to visual generation by flattening 2D images, pixels, or discrete visual tokens into 1D sequences and modeling them with standard autoregressive Transformers, following a raster or otherwise fixed token order (Chen et al., 2020; Ramesh et al., 2021; Esser et al., 2021; Yu et al., 2022; Sun et al., 2024). While conceptually simple and scalable, this direct transfer is not perfectly aligned with the spatial nature of images. Unlike text, where the generation order often follows a natural linguistic progression, images contain dense 2D structures and long-range spatial dependencies that do not naturally conform to a single left-to-right sequence. As a result, fixed-order AR generation can easily underutilize important structural cues, leading to visible artifacts such as row-wise or column-wise misalignment, distorted object layouts, and weakened global coherence. To alleviate these limitations, subsequent works explored more flexible generation schemes, including masked generative modeling that predicts randomly masked tokens or iteratively refines token sets (Chang et al., 2022, 2023; Li et al., 2023, 2024c; Kim et al., 2025b), as well as scale-wise au-

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

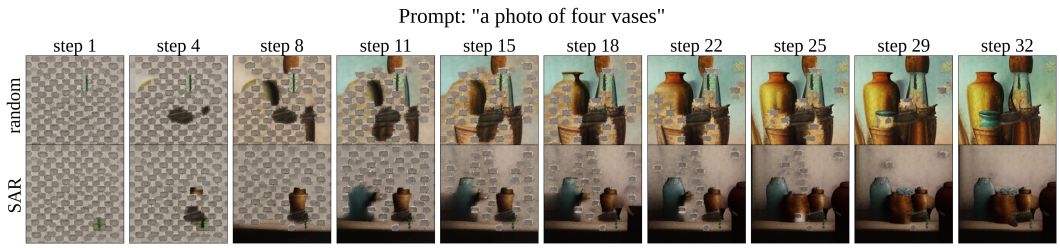


Figure 1: ***Generation order shapes reasoning-like behaviors*** in autoregressive image generation. With random-order decoding (top), intermediate predictions remain scattered and incomplete, weakening feedback for counting (e.g., "four vases") and often causing over-generation. With context-aware location selection (bottom), the same model and seed tend to complete objects more coherently (roughly one-by-one) and stop adding new ones once the count is met.

toregressive generation that predicts visual tokens progressively from coarse to fine resolutions (Tian et al., 2024b; Ma et al., 2024; Han et al., 2025b), which reduce the rigidity of raster decoding and improve visual fidelity and structural stability.

Nonetheless, although random-order and scale-wise generation alleviate the structural distortion, the reasoning logic behind patch-level generation remains largely overlooked. For example, ***when humans draw an image patch by patch, they can usually decide where to draw next based on the existing context.*** This allows them to build a more coherent spatial organization and progressively reason over the partially completed content. In contrast, masked AR models generate patches in a random order, while scale-wise AR models generate all tokens within a given spatial scale in one step. Both strategies can easily miss an important intermediate reasoning process. We illustrate this issue with an intuitive example in Fig. 1, where the model is asked to generate *four vases*. In Case 2, when the model is allowed to determine the next prediction location according to context, an emerging reasoning behavior can be observed. The model generates the vases in an organized manner, roughly one by one, and obtains self-feedback during the generation process. For instance, before step 22, it has already generated three vases. In the following steps, the model appears to recognize that one vase is still missing, and completes the fourth vase by step 25. After the required count is satisfied, the remaining steps are mainly used to fill in the background. In contrast, in Case 1, the random-order AR model selects prediction locations randomly at each step. As a result, most objects remain incomplete during the intermediate generation process, making it difficult for the model to receive reliable feedback about the object count. Consequently, it keeps generating new vase-like regions and fails to align with the required number.

Motivated by this observation, we propose **Saccade AR (SAR)**[1], a simple yet surprisingly effective context-aware decoding strategy for AR image generation. In detail, at each decoding step, an AR model typically produces candidate predictions for all editable token positions, after which existing methods commit tokens according to a random, predefined, or scale-wise schedule. Instead, SAR ranks candidate token locations by their semantic relevance to the global context. Specifically, we measure the distance between each candidate token representation and the global conditioning signal, such as the class token in class-conditional generation or text embeddings in text-to-image generation, and prioritize locations from semantically close to semantically distant. In this way, SAR first generates regions that are strongly related to the global intent, and then fills in less critical details. SAR is training-free, lightweight, and broadly applicable. It can be plugged into pretrained AR image models at inference time without changing model parameters, and it introduces almost no additional computation because it only reorders which candidate predictions are committed. Moreover, SAR applies to different AR paradigms, including masked autoregressive models (MAR (Li et al., 2024c)) and scale-wise autoregressive models (VAR (Tian et al., 2024b)). Importantly, SAR does not change the model architecture or supervision; it changes the generation trajectory. Our central thesis is that ***this trajectory matters***: when the model generates in a context-aware order, it can better exploit intermediate visual feedback and exhibit stronger structural robustness and reasoning-related instruction following.

[1]We call our method *Saccade* AR because it is inspired by saccadic eye movements in the human visual system, which actively choose where to fixate based on context when interpreting a visual scene.

Extensive experiments validate this hypothesis. On text-to-image benchmarks such as GenEval (Ghosh et al., 2023) and DPG-Bench (Hu et al., 2024), SAR consistently improves pre-trained AR generators including MaskGen (Kim et al., 2025b) and Infinity (Han et al., 2025b), with especially clear gains on reasoning-related dimensions such as counting, relative positioning, and compositional instruction following. To further evaluate reasoning ability, we construct a long-prompt text-to-image benchmark involving complex combinations of object counts, spatial relations, and fine-grained attribute assignments. On this benchmark, SAR brings even larger improvements, supporting our claim that generation order is a critical factor behind the reasoning limitations of existing AR image models.

2 Related Work

Autoregressive models in language and multimodal generation. Recent advances in large language models (LLMs) (Brown et al., 2020; Touvron et al., 2023a,b; Gemini Team, Google, 2023; OpenAI, 2024; Yang et al., 2024) have made autoregressive (AR) models central to modern generative modeling, due to their flexibility and wide use in predicting each next token from prior tokens. First developed for text, AR models are now also widely used for visual generation (Chen et al., 2020; Esser et al., 2021; Sun et al., 2024; Tian et al., 2024a; Li et al., 2024b), where each image is treated as a sequence of discrete tokens. The same idea has further led to unified vision–language models that handle both multimodal understanding and generation, such as Emu3 (Emu3 Team, BAAI, 2024), LaVIT (Jin et al., 2023),and VILA-U (Wu et al., 2024). In these models, one shared AR backbone reads and writes mixed text and image tokens.

Autoregressive image generation with raster-scan token order. Early research on AR visual modeling adopts language model architectures to generate images pixel by pixel (Chen et al., 2020). With the help of image tokenizers (Van Den Oord et al., 2017; Esser et al., 2021), later AR models produce images as discrete token sequences in a next-token manner, where each image is flattened into a 1D sequence and predicted by a decoder-only transformer, similar to language modeling. This pipeline has been scaled up to large-scale text-conditioned generation, such as DALL-E (Ramesh et al., 2021), CogView (Ding et al., 2021, 2022), Make-A-Scene (Gafni et al., 2022), Parti (Yu et al., 2022), LlamaGen (Sun et al., 2024), and Lumina-mGPT (Liu et al., 2024), with parallel extensions to text-to-video models such as VideoPoet (Kondratyuk et al., 2023). However, these methods produce image tokens in raster-scan order, which does not fully use the 2D structure of natural images, leading to structural artifacts and weakened coherence under complex semantic instructions.

Improved generation orders for autoregressive image synthesis. Beyond raster-scan generation, recent works explore two complementary directions. *Mask-based methods* predict multiple tokens per step in a non-fixed order: MaskGIT (Chang et al., 2022) predicts groups of tokens jointly at each step, cutting the inference cost; Muse (Chang et al., 2023) extends this idea to text-to-image generation; MAGE (Li et al., 2023) unifies representation learning and synthesis under the same masked-token objective. MAR (Li et al., 2024b) brings masked prediction to continuous-valued tokens via a small diffusion head, replacing the discrete codebook, and MaskGen (Kim et al., 2025a) compresses images into compact 1D tokens for masked text-to-image generation. *Scale-wise methods* replace the spatial axis with a coarse-to-fine scale axis: VAR (Tian et al., 2024a) sums predictions across scales so the inference step count equals the number of scales, with follow-ups in STAR (Ma et al., 2024) and Infinity (Han et al., 2025a). While both directions alleviate the rigidity of raster-scan, the resulting order is still either fixed *a priori* (a predetermined scale schedule, as in VAR) or determined *stochastically* (random or confidence-based unmasking, as in MaskGIT and MAR)—no existing method dynamically chooses the next step based on the semantic context already produced. This motivates our *context-aware* generation order, in which the model itself decides where to attend and generate next, analogous to the saccadic behaviour of the human visual system.

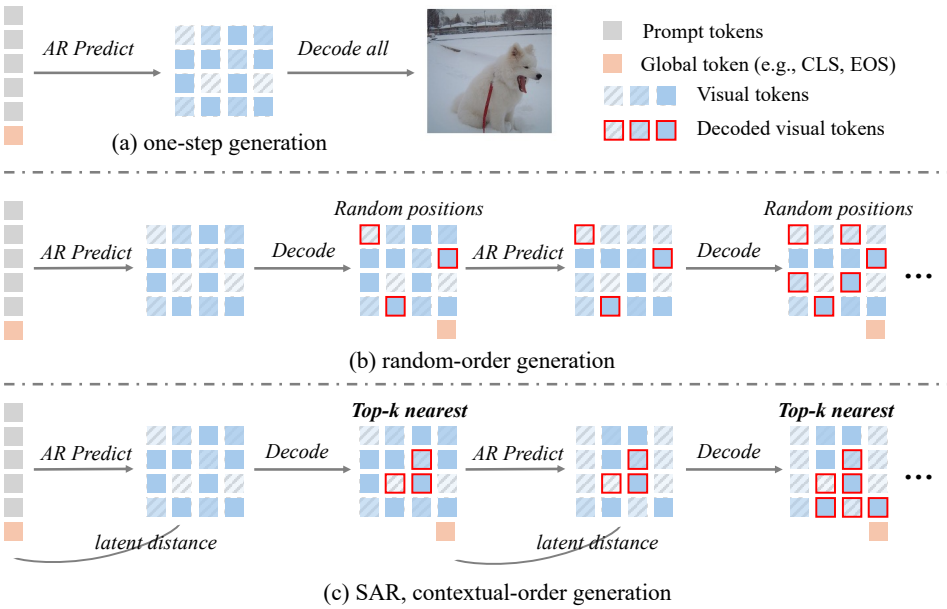


Figure 2: Comparison to one-step and random-order generation paradigms. Our SAR model decides the generation order based on each token's latent distance to a global token. ***This is only a minimal change for image AR models, but we find it crucial for the AR model's reasoning behavior.***

3 Method

3.1 Image Autoregressive Models

Modern image autoregressive models can be grouped into two families that differ in the unit of generation. *Patch-wise* models tokenize the image into a flat sequence of spatial (or 1D-compressed) latent tokens and predict them one or a few at a time. *Scale-wise* models predict the image at progressively finer resolutions, generating a coarse-to-fine pyramid of tokens. We describe the patch-wise scale-wise formulations below, and then explain how SAR adapt with them.

Patch-wise masked autoregressive models. We adopt the masked-autoregressive (MAR) formulation (Li et al., 2024b), encoding an image x into N continuous latent tokens $z = \{z_i\}_{i=1}^{N}$ and factorizing the conditional distribution over a generation order π:

$$p_\theta(z \mid c) \;=\; \prod_{t=1}^{T} p_\theta(z_{S_t} \mid z_{S_{<t}},\, c)\,, \tag{1}$$

Here c is a conditioning signal, and $\{S_t\}_{t=1}^{T}$ partitions token indices according to π. At step t, the model predicts S_t in parallel given $z_{S_{<t}}$, unmasking $|S_t|$ tokens with the cosine schedule above. Each per-token conditional is implemented by an MAE-style transformer decoder with a lightweight denoising-diffusion head, and we apply classifier-free guidance (CFG) with a linear schedule across AR steps at inference.

Scale-wise autoregressive models. We also consider scale-wise AR as in VAR (Tian et al., 2024a) and Infinity (Han et al., 2025a), which generates a coarse-to-fine pyramid $z = \{z^{(k)}\}_{k=1}^{K}$ (with N_k tokens at scale k):

$$p_\theta(z \mid c) \;=\; \prod_{k=1}^{K} p_\theta\Big(z^{(k)} \,\Big|\, z^{(<k)},\, c\Big)\,, \tag{2}$$

At scale k, the model predicts the next finer tokens conditioned on coarser scales $z^{(<k)}$ and c, progressively refining details. Instantiations differ across models: VAR uses discrete next-scale tokens, while Infinity predicts bits with a multi-scale residual tokenizer.

3.2 Saccade Autoregressive Generation

The order π in Eq. 1 is typically fixed (e.g., raster/zigzag) or sampled uniformly at random (Li et al., 2024b). Neither leverages image spatial semantics. Inspired by human saccades, we instead choose the order adaptively at inference time, generating the tokens most relevant to the conditioning first, with saliency decided by the model rather than a hand-designed scan. Let $h_i^{(t)} \in \mathbb{R}^d$ denote the model-specific candidate representation used for ranking at masked position $i \in \mathcal{M}_t$ and AR step t. Let $\mathcal{A}^{(t)} \subset \mathbb{R}^d$ be conditioning *anchors* in the same embedding space: the conditioning tokens visible before any image token is generated (class tokens for class-conditional, prompt tokens for text-conditional). Model-specific details are in Sec. 4. Because the generation pipeline produces candidate representations for *all* masked positions at each step, we can rank positions on the fly. We score each still-masked position by its average cosine similarity to anchors,

$$s_i^{(t)} = \frac{1}{|\mathcal{A}^{(t)}|} \sum_{a \in \mathcal{A}^{(t)}} \frac{\langle h_i^{(t)}, a \rangle}{\|h_i^{(t)}\| \, \|a\|}, \qquad i \in \mathcal{M}_t, \tag{3}$$

then unmask the top-$|S_t|$ positions (same cosine schedule as Eq. 1). Recomputing $h^{(t)}$ each step yields an adaptive, context-dependent order. SAR changes only inference-time ranking and adds a single similarity computation per (masked token, anchor, step).

MAR feature selection. For MAR, position selection is performed after token-wise diffusion denoising, rather than directly on the deterministic masked-prediction features. At each step, the model first predicts candidates for unresolved positions from the current visible context, and then samples denoised candidate tokens before SAR selects which positions to commit. Diffusion sampling introduces seed-dependent variation, so the first selected position need not be fixed for a given class condition. After each commitment, the updated partial image conditions the next round of candidate generation and ranking.

SAR for scale-wise generation. Scale-wise AR (Eq. 2) generates all N_k tokens at a scale in one shot, so there is no intra-scale order. We introduce an order by decoding only the final scale K over $M{>}1$ steps (scales $1 \dots K{-}1$ unchanged), since the last scale has the highest spatial resolution and contains most fine details. We split the N_K final-scale positions into M equal groups and commit one group per step. At step m, committed positions use their decoded embeddings and the rest keep the coarse features from scales $1 \dots K{-}1$; a non-causal pass produces fresh hidden states $h_i^{(m)}$ for all uncommitted positions. We rank positions by squared L2 distance to an anchor a,

$$s_i^{(m)} = \left\| h_i^{(m)} - a \right\|_2^2, \tag{4}$$

and commit the N_K/M positions with smallest scores. We reuse the model's existing conditioning embedding as a (class embedding c_{class} or pooled text embedding c_{pool}), computed once and fixed across steps, so this adds essentially no extra overhead.

Overall, scale-wise SAR is an inference-time re-ranking on top of M-step last-scale decoding; it requires no parameter updates and adds one similarity per uncommitted position per step. We instantiate it on VAR (Tian et al., 2024a) (class-conditional; Appendix B) and Infinity (Han et al., 2025a) (text-to-image; Sec. 4.2). Table 7 compares inference-time orders on shared checkpoints. Separately, Table 8 studies VAR retrained using SAR ordering at the last scale, followed by evaluation under different inference orders.

4 Experiments

In this section, we evaluate SAR across both patch-wise and scale-wise AR. Sec. 4.1 qualitatively visualizes the intermediate decoded image at every AR step under random vs. SAR; Sec. 4.2 reports our main text-to-image results on MaskGen and Infinity, covering GenEval, DPG-Bench, MJHQ-30K FID, and a complex long-prompt reasoning benchmark we construct (Sec. 4.2.1); Sec. 4.3 presents ablations on alternative orders and on the VAR retraining. We also provide class-conditional ImageNet results on MAR and VAR in Appendix B.

4.1 Qualitative Analysis

Before the quantitative results we build intuition for *how* SAR helps by visualizing the intermediate decoded image at every AR step under random vs. SAR, with the checkpoint and prompt held fixed. We use one representative model per AR family: Infinity-8B (scale-wise) and MaskGen-XL (patch-wise). Implementation details are provided in Sec. 4.2. The two realize SAR's improvement through *different* reasoning behaviors that converge on the same effect.

Infinity: outline first, details later. On Infinity-8B (Fig. 3), SAR commits to the prompt-correct *global outline* (count, object identity, and overall geometry) within the first few last-scale steps, and uses subsequent steps only to fill in textures. Random ordering scatters partial textures before any outline is established and often fails to recover those structural decisions by the final step.

MaskGen: one object at a time. On MaskGen-XL (Fig. 4), SAR instead commits one full object at a time, moving on only once the previous object is complete; counting, spatial relations, and compositional hierarchy all fall out of this sequential commitment. Random ordering grows all parts concurrently and resolves identity and layout only in the last few steps.

The two families realize SAR through complementary mechanisms (*outline-to-detail* vs. *object-by-object*), but in both cases SAR's prompt-grounded ordering turns AR decoding into a sequence of legible, prompt-aligned commitments rather than a soup of half-formed concepts. This implicit visual planning emerges *for free*, and the quantitative results across all 8 models in Sec. 4.2 and Appendix B confirm that this behavioral change translates into consistent benchmark gains.

4.2 Text-to-image generation

Setup. We evaluate text-to-image SAR on two AR families that span the dominant design choices in the literature: **MaskGen** (Kim et al., 2025a) (patch-wise; MaskGen-L and MaskGen-XL) and **Infinity** (Han et al., 2025a) (scale-wise; Infinity-2B and Infinity-8B). MaskGen takes the SAR anchor to be the singleton $\{W\, c_{\text{pool}}\}$, with W the same linear projection MaskGen already uses to inject pooled text conditioning; Infinity takes the SAR anchor to be $a = c_{\text{pool}}$, the attentive-pooled text embedding already consumed by AdaLN. We use $T{=}32$ AR steps for MaskGen and $M{=}16$ equal groups at the last scale for Infinity. All other implementation details follow the original recipes and are listed in Appendix A.

Evaluation. We use three complementary benchmarks: GenEval (Ghosh et al., 2023) for compositional capability, with 4 samples per prompt and the prompt-level pass criterion; DPG-Bench (Hu et al., 2024) for fine-grained prompt alignment, scored by the mPLUG-VQA evaluator on a $2{\times}2$ image grid; and MJHQ-30K FID (Li et al., 2024a) for distribution-level aesthetic quality on a 30K-prompt subset. Following the original Infinity protocol, GenEval is run with prompt rewriting on Infinity; all other settings use raw prompts. All three are averaged over 3 seeds.

Results. Tables 2 and 3 show that the inference-only SAR change improves compositional capability at every scale across both paradigms. On GenEval, the Overall pass rate rises consistently, with the largest gains on sub-categories that require global consistency such as *Counting*, *Colors*, and *Position*. On DPG-Bench, the per-category L1 breakdown is mixed but the aggregate Overall score also improves at every scale. On MJHQ-30K FID (Table 4), SAR is statistically tied with random at every scale, indicating that the compositional gains do not come at the price of distribution-level aesthetic quality. Together these results demonstrate that a single inference-time intervention generalizes across both patch-wise and scale-wise text-to-image AR. For the scale-wise setting, Appendix C additionally studies SAR-order last-scale retraining of VAR and compares inference orders on the resulting checkpoints.

4.2.1 Complex long-prompt benchmark

To complement GenEval and DPG-Bench, we further build a complex long-prompt benchmark to better evaluate reasoning-intensive instruction following. The benchmark contains 2,000 prompts with dense compositional constraints, including object counts, spatial relations, attribute binding, object existence, compositional ordering, and view consistency. A typical prompt is: "A vertical stack of four different items: a wooden donut at the bottom, a metallic sphere in the middle, and a

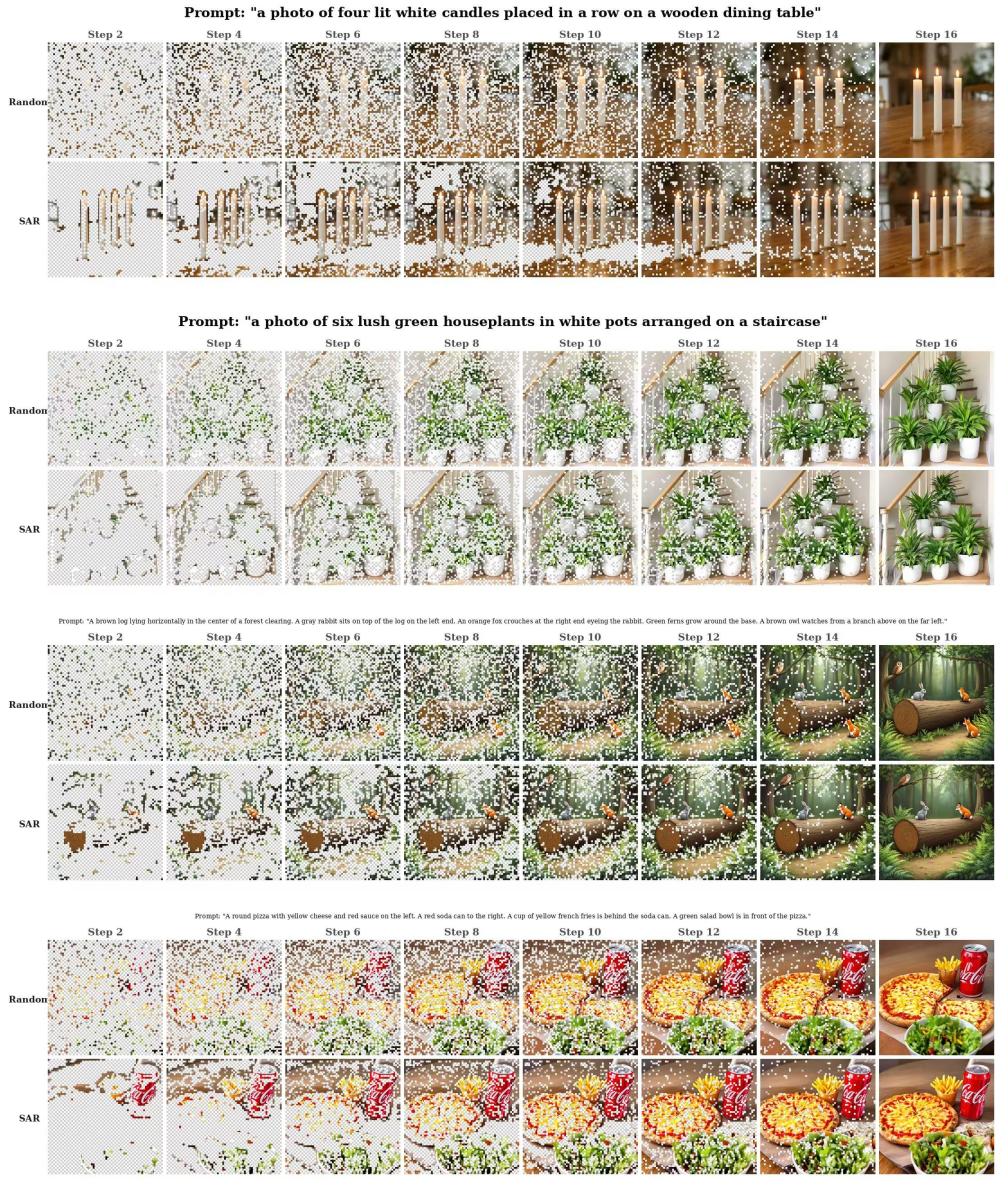


Figure 3: Infinity-8B: *outline-to-detail* reasoning under SAR. In each strip, the top row is random ordering and the bottom row is SAR; columns are AR steps. Within the first few SAR steps the global outline already encodes the correct count (four candles, six houseplants), the correct object set with no hallucinated extras (fox+rabbit), and the correct global geometry (circular pizza). Subsequent steps fill in textures without changing those structural decisions. Random ordering instead grows partial textures everywhere and frequently fails to recover count, identity, or shape by the final step.

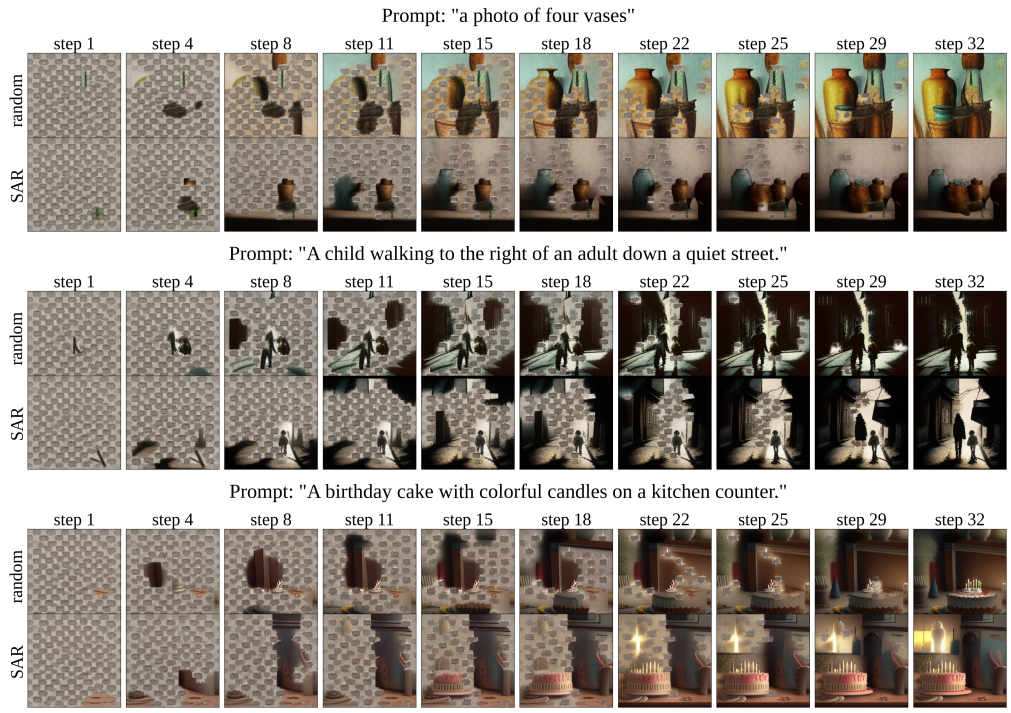


Figure 4: MaskGen-XL: *object-by-object* reasoning under SAR. Top to bottom: "four vases", "a child walking to the right of an adult down a quiet street", and "a birthday cake with colorful candles on a kitchen counter". SAR completes one object before starting the next (counting), commits one entity before introducing the other (spatial relation), and follows scene $\rightarrow$ main object $\rightarrow$ parts (compositional hierarchy); random ordering grows everything concurrently and only resolves identity and layout in the last few steps.

Table 1: Results on our complex long-prompt benchmark. Scores are judged by Gemini 2.5 Pro on a 1–5 scale.

Model	Order	Cnt.	Spat.	Attr.	Exist.	Ord.	View	Avg.
MaskGen-XL	random	2.55	2.24	2.39	2.51	2.32	2.51	2.42
	SAR	**3.08**	**2.78**	**2.94**	**3.01**	**2.82**	**2.89**	**2.92**
Infinity-2B	random	2.76	2.43	2.64	2.72	2.49	2.68	2.62
	SAR	**3.27**	**2.98**	**3.16**	**3.22**	**2.99**	**3.10**	**3.12**
Infinity-8B	random	3.09	2.74	2.94	3.02	2.82	2.91	2.92
	SAR	**3.61**	**3.31**	**3.45**	**3.49**	**3.30**	**3.36**	**3.42**

miniature cactus on the very top. This stack is placed inside a hollow red ring. To the front-left of the stack is a spilled glass of milk, and to the back-right is a closed leather book. Cinematic top-down 45-degree angle view."

For evaluation, we generate images from each prompt using different AR models and decoding orders. We then use Gemini 2.5 Pro (Gemini Team, Google, 2025) as a VLM judge. For each image, we construct 10 question-answer pairs from the prompt, with ground-truth answers directly extracted from the prompt constraints. Gemini is asked to score the match between the generated image and each question on a 1–5 scale. We report the average scores under six reasoning-related categories.

Table 1 shows that SAR consistently improves complex instruction following across all three evaluated AR models, raising the average score by roughly $+0.5$ on each. The largest gains tend to appear on categories that require global consistency (counting, spatial relation, and compositional ordering) where random-order decoding fails to maintain reliable intermediate feedback about how

Table 2: GenEval pass rate (%, ↑) for both AR paradigms. Best per (model, metric) pair in bold.

Model	Method	S.Obj	T.Obj	Count	Colors	Pos.	C.Attr	Overall
MaskGen-L	Random	**100.00**	54.55	40.00	85.11	18.00	26.00	53.94
	SAR	**100.00**	**56.57**	**50.00**	**89.36**	**20.00**	**29.00**	**57.49**
MaskGen-XL	Random	**98.75**	58.59	43.75	87.23	21.00	28.00	56.22
	SAR	**98.75**	**64.65**	**46.25**	**88.30**	**25.00**	**37.00**	**59.99**
Infinity-2B	Random	**100.00**	82.62	64.38	82.77	**42.10**	58.88	71.79
	SAR	**100.00**	**83.34**	**65.27**	**85.42**	41.00	**60.38**	**72.57**
Infinity-8B	Random	**100.00**	**92.04**	68.69	86.03	**66.76**	**70.24**	80.63
	SAR	**100.00**	91.76	**73.21**	**86.74**	**66.76**	69.57	**81.34**

Table 3: DPG-Bench L1 category breakdown (%, ↑) for both AR paradigms. Best per (model, metric) pair in bold.

Model	Method	Entity	Attribute	Relation	Global	Other	Overall
MaskGen-L	Random	**81.82**	80.04	80.91	76.12	81.08	70.11
	SAR	78.30	**80.84**	**82.91**	**82.75**	**81.70**	**70.51**
MaskGen-XL	Random	79.33	**80.98**	81.63	80.81	80.52	71.00
	SAR	**81.53**	79.82	**82.83**	**82.81**	**81.08**	**71.57**
Infinity-2B	Random	89.14	86.19	92.50	**83.18**	82.67	82.71
	SAR	**89.24**	**86.23**	**92.51**	82.88	**82.93**	**82.76**
Infinity-8B	Random	91.38	88.75	94.24	83.99	**86.80**	86.09
	SAR	**91.45**	**88.87**	**94.25**	**85.01**	86.67	**86.27**

many objects have already been generated or where they should be placed. This supports our hypothesis that generation order is not merely a low-level sampling detail but directly affects whether an AR image model can use partially generated content as feedback and reason over the remaining constraints during generation.

4.3 Ablation Study

To check that the gains of SAR are not a trivial side-effect of merely *abandoning* the random baseline, we compare it against two deterministic alternatives: *Sequential*, which generates tokens one by one in their natural index order, and *Inside-out*, which starts from the middle token and expands outward symmetrically. We evaluate them on one representative model from each AR family: MaskGen-XL (patch-wise) and Infinity-8B (scale-wise). Table 5 reports GenEval and DPG-Bench Overall scores under all four orders. SAR consistently outperforms all three alternative orderings on both benchmarks and on both models, indicating that what matters is conditioning the choice of next position on the prompt itself, rather than simply imposing a structural pattern.

Controlled comparison with confidence scheduling. MaskGIT (Chang et al., 2022) selects positions using prediction confidence, whereas SAR explicitly ranks candidate representations by their alignment with the conditioning anchor. Both rules can depend on the current context; the distinction is the scoring criterion, not dynamic selection alone. We replace the confidence selector in a pretrained MaskGIT-Base checkpoint with SAR, keeping the checkpoint, 12-step cosine schedule, classifier-free guidance setting, and 50K class-conditional ImageNet 256×256 sample count fixed, without retraining. Table 6 shows lower FID and higher recall under SAR, isolating the benefit of the selection criterion.

Inference cost and matched-budget comparisons. We measure VAR end-to-end generation latency on one NVIDIA H100 80GB GPU with BF16 and batch size 1. After 20 warm-up generations, we time 1,000 ImageNet images, including VAE decoding but excluding model loading and image

Table 4: MJHQ-30K FID (↓) for both AR paradigms. Across all four scales, SAR is statistically tied with the random-order baseline, indicating that context-aware ordering does not change the distribution-level aesthetic quality of the generated images.

	Patch-wise (MaskGen)		Scale-wise (Infinity)	
Order	MaskGen-L	MaskGen-XL	Infinity-2B	Infinity-8B
Random	6.79	6.75	10.867	9.332
SAR	6.79	6.74	10.970	9.317

Table 5: Ablation on the inference-time generation order, on one representative model from each AR family. "Sequential" generates tokens one by one in their natural index order; "Inside-out" starts from the middle token and expands outward. Higher is better for all columns; best per column in bold.

	MaskGen-XL		Infinity-8B	
Order	GenEval (↑)	DPG-Bench (↑)	GenEval (↑)	DPG-Bench (↑)
Random	0.55	71.00	0.804	86.05
Sequential	0.57	71.00	0.799	86.01
Inside-out	0.55	71.30	0.808	86.25
SAR (ours)	**0.58**	**71.57**	**0.815**	**86.36**

Table 6: Controlled MaskGIT-Base comparison: only the position-selection criterion changes.

Selection rule	FID ↓	Recall ↑
Confidence (original)	6.48	0.41
SAR	**5.95**	**0.43**

saving. Random and SAR use the same retrained checkpoint and identical four-step last-scale decoding, differing only in position selection. Table 9 in Appendix D reports mean latency and population standard deviation alongside generation quality. Across all four VAR sizes, Random and SAR have identical mean latency at the reported precision, although their timing variability can differ. This indicates no measurable additional overhead from SAR's ranking in this setup, rather than zero computational cost. The increase relative to one-step decoding comes from the shared multi-step last-scale procedure; the matched four-step comparison isolates the effect of the ordering rule.

5 Conclusion

We introduced **SAR**, a context-aware generation order that, at every AR step, ranks the still-uncommitted positions by similarity to a prompt-derived anchor (the conditioning embedding the base model already maintains) and commits the most relevant ones first; SAR introduces no new parameters and runs purely at inference. Across 8 checkpoints spanning both AR paradigms (MAR, MaskGen, VAR, Infinity) and four standard benchmarks plus our complex long-prompt benchmark, SAR consistently improves prompt alignment and compositional reasoning while preserving distribution-level aesthetic quality, and qualitatively elicits two complementary reasoning behaviors: outline-first decoding in scale-wise models and object-by-object decoding in patch-wise models. The fact that this implicit visual planning emerges for free from the model's own representations suggests that generation order is a critical and underexplored axis of AR image modeling, with richer anchors and combinations with aesthetic-aware criteria as natural directions for further gains.

Acknowledgments and Disclosure of Funding

Authors from Johns Hopkins University are supported by ONR grants N000142512132 and N000142412696.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 3, 9

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. In *International Conference on Machine Learning*, 2023. 1, 3

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. ImageGPT: Generative pretraining from pixels. In *International Conference on Machine Learning*, 2020. 1, 3

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 14, 22

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 14

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. CogView: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34:19822–19835, 2021. 3

Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. CogView2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*, 35:16890–16902, 2022. 3

Emu3 Team, BAAI. Emu3: Next token prediction is all you need. https://baai-solution.ks3-cn-beijing.ksyuncs.com/emu3/Emu3-tech-report.pdf, 2024. 3

Patrick Esser, Robin Rombach, and Björn Ommer. VQGAN: Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 3

Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *European Conference on Computer Vision*, pages 89–106. Springer, 2022. 3

Gemini Team, Google. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 3

Gemini Team, Google. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 8, 22, 23

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023. 3, 6, 22

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a. 3, 4, 5, 6, 14, 22

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025b. 2, 3

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 14

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024. 3, 6, 22

Yang Jin, Kun Xu, Kun Xu, Liwei Chen, Chao Liao, Jianchao Tan, Yadong Mu, et al. Unified language-vision pretraining in LLM with dynamic discrete visual tokenization. *arXiv preprint arXiv:2309.04669*, 2023. 3

Dongwon Kim, Ju He, Qihang Yu, Chenglin Yang, Xiaohui Shen, Suha Kwak, and Liang-Chieh Chen. Democratizing text-to-image masked generative models with compact text-aware one-dimensional tokens. *arXiv preprint arXiv:2501.07730*, 2025a. 3, 6, 14, 22

Dongwon Kim, Ju He, Qihang Yu, Chenglin Yang, Xiaohui Shen, Suha Kwak, and Liang-Chieh Chen. MaskGen: Democratizing text-to-image masked generative models with compact text-aware one-dimensional tokens. *arXiv preprint arXiv:2501.07730*, 2025b. 1, 3

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Rachel Hornung, Hartwig Adam, Hassan Akbari, Yair Alon, Vighnesh Birodkar, et al. VideoPoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023. 3

Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2. 5: Three insights towards enhancing aesthetic quality in text-to-image generation. *arXiv preprint arXiv:2402.17245*, 2024a. 6, 22

Tianhong Li, Huiwen Chang, Shlok Kumar Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. MAGE: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1, 3

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *arXiv preprint arXiv:2406.11838*, 2024b. 3, 4, 5, 14, 22

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. MAR: Autoregressive image generation without vector quantization. In *Advances in Neural Information Processing Systems*, 2024c. 1, 2

Dongyang Liu, Shitian Zhao, Le Zhuo, Weifeng Lin, Yu Qiao, Hongsheng Li, and Peng Gao. Lumina-mGPT: Illuminate flexible photorealistic text-to-image generation with multimodal generative pretraining. *arXiv preprint arXiv:2408.02657*, 2024. 3

Xiaoxiao Ma, Mohan Zhou, Tao Liang, Yalong Bai, Tiejun Zhao, Biye Li, Huaian Chen, and Yi Jin. STAR: Scale-wise text-conditioned autoregressive image generation. *arXiv preprint arXiv:2406.10797*, 2024. 2, 3

OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, 2024. 3

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. DALL-E: Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 2021. 1, 3

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. LlamaGen: Autoregressive model beats diffusion. *arXiv preprint arXiv:2406.06525*, 2024. 1, 3

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *arXiv preprint arXiv:2404.02905*, 2024a. 3, 4, 5, 14, 22

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. VAR: Visual autoregressive modeling via next-scale prediction. In *Advances in Neural Information Processing Systems*, 2024b. 2

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a. 3

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b. 3

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017. 3

Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. VILA-U: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024. 3

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024. 3

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Parti: Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 1, 3

A Implementation Details

Code for the VAR and Infinity implementations is publicly available at https://github.com/Hpesojyjx/Saccades-AR.

A.1 Text-to-image generation (Sec. 4.2)

MaskGen. We use the public MaskGen-L and MaskGen-XL checkpoints (Kim et al., 2025a), both built on the KL variant of the TATiTok 1D image tokenizer that compresses an image into $N{=}32$ continuous latent tokens. A frozen CLIP ViT-L/14-336 text encoder produces a per-token text feature sequence and a pooled embedding c_{pool}; the linear projection W used to form the SAR anchor is the same one MaskGen already uses to inject pooled text conditioning into its decoder. We sample with $T{=}32$ AR steps, CFG scale 3.0 with a linear schedule, and an aesthetic micro-conditioning score of 6.5, matching the original MaskGen recipe.

Infinity. We use Infinity-2B and Infinity-8B (Han et al., 2025a), both built on the BSQ-VAE tokenizer with $K{=}13$ coarse-to-fine scales at $1024{\times}1024$ resolution; the final scale alone contains 4096 spatial positions and accounts for the bulk of the token budget. A frozen FLAN-T5-XL text encoder produces a per-token feature sequence consumed by cross-attention and a learned attentive-pooled embedding c_{pool} used for AdaLN modulation. We run multi-step inference at the final scale only, with $M{=}16$ equal groups, leaving scales 1 through $K{-}1$ unchanged. CFG is 4.0 for Infinity-2B and 3.0 for Infinity-8B with sampling temperature $\tau{=}1$, matching the original Infinity recipe.

A.2 Class-conditional generation (Appendix B)

MAR. We use MAR-Base (208M) and MAR-Large (479M) (Li et al., 2024b). The image is encoded by a KL-regularized 2D VAE into $N{=}256$ continuous spatial tokens of dimension 16, and the class label is injected as a buffer of class-token embeddings prepended to the decoder input. We sample with $T{=}256$ AR steps, 100 diffusion sampling steps inside the per-token diffusion head, and CFG with a linear schedule (CFG scale 2.9 for MAR-Base, 3.0 for MAR-Large), all matching the original MAR recipe.

VAR. We use VAR-$d16$ (310M) and VAR-$d20$ (600M) (Tian et al., 2024a). Each image is encoded by a multi-scale residual VQ tokenizer into a coarse-to-fine pyramid of $K{=}10$ token maps with final resolution $16{\times}16$ at $256{\times}256$ output, and the class label is injected as a class embedding $c_{\text{class}}{=}\text{ClassEmb}(y)$ driving AdaLN modulation. We run last-scale multi-step inference with $M{=}4$ equal groups, leaving scales 1 through $K{-}1$ unchanged, and adopt the CFG schedule and sampling temperature of the original VAR recipe.

B Class-conditional Generation Results

Setup. We evaluate class-conditional SAR on two AR families: **MAR** (Li et al., 2024b) (patch-wise; MAR-Base and MAR-Large) and **VAR** (Tian et al., 2024a) (scale-wise; VAR-$d16$ and VAR-$d20$). MAR takes the SAR anchor to be the decoder's representation of the class-token buffer prepended to its decoder input; VAR takes the SAR anchor to be $a = c_{\text{class}}{=}\text{ClassEmb}(y)$, the class embedding already consumed by AdaLN modulation. We use $T{=}256$ AR steps for MAR and $M{=}4$ equal groups at the last scale for VAR. The VAR multi-step inference is enabled by an order-agnostic random-order retraining of each VAR variant; this retraining is shared by the random and SAR rows of Table 7 and is itself ablated in Appendix C. All other implementation details follow the original recipes and are listed in Appendix A.

Evaluation. We evaluate on class-conditional ImageNet-1K (Deng et al., 2009) at $256{\times}256$, reporting Fréchet Inception Distance (FID) (Heusel et al., 2017) computed against the ADM reference statistics (Dhariwal and Nichol, 2021).

Results. Table 7 reports class-conditional FID. SAR yields a consistent improvement over the random-order baseline at every model size and across both paradigms, despite changing only the

Table 7: Class-conditional FID (↓) on ImageNet 256×256 across both AR paradigms. Within each column, both rows share the same checkpoint (the VAR rows use a retrained checkpoint with $M{=}4$ multi-step inference at the last scale) and the same sampling configuration; only the inference-time generation order differs.

	Patch-wise (MAR)		Scale-wise (VAR)	
Order	MAR-Base	MAR-Large	VAR-$d16$	VAR-$d20$
Random	2.29	1.78	3.56	2.77
SAR (class similarity)	**2.21**	**1.75**	**3.48**	**2.73**

Table 8: Results of retraining VAR models with different generation orders. For each VAR model, we apply the SAR order at the last scale during retraining. The superscript * indicates the original setting where both training and evaluation generate all tokens within each scale in one step.

Model	Order	FID ↓	IS ↑	Precision ↑	Recall ↑
VAR-$d16$	1-step*	3.30	274.4	0.84	0.51
	1-step	3.42	268.5	0.83	0.49
	random	3.24	278.2	0.84	0.52
	SAR	**3.05**	**286.2**	**0.86**	**0.54**
VAR-$d20$	1-step*	2.57	302.6	0.83	0.56
	1-step	2.68	297.0	0.82	0.54
	random	2.53	304.8	0.84	0.57
	SAR	**2.35**	**314.5**	**0.85**	**0.59**
VAR-$d24$	1-step*	2.09	312.9	0.82	0.59
	1-step	2.20	307.4	0.81	0.57
	random	2.06	315.4	0.83	0.60
	SAR	**1.88**	**325.3**	**0.84**	**0.62**
VAR-$d30$	1-step*	1.92	323.1	0.82	0.59
	1-step	2.03	317.2	0.81	0.57
	random	1.90	325.9	0.82	0.60
	SAR	**1.74**	**335.5**	**0.84**	**0.62**

inference-time order. Ranking tokens by similarity to the class embedding provides a more informative early context than uniform random selection.

C VAR Retraining Ablation

Table 8 reports a separate training-time experiment from the inference-order comparison in Table 7. Here, VAR-$d16$, $d20$, $d24$, and $d30$ are retrained using SAR ordering at the last scale. The 1-step* row reports the original VAR setting, in which training and inference generate all tokens within each scale in one step. The remaining rows evaluate the retrained checkpoint under different inference procedures: 1-step generates all tokens within each scale at once, while random and SAR use four-step last-scale decoding with their respective position-selection rules. The 1-step results remain close to the original baseline. Under the same retrained checkpoint and four-step inference budget, SAR improves FID, IS, precision, and recall over random ordering at every model size. Thus, the random-versus-SAR comparison measures the effect of inference order after SAR-order retraining; improvements relative to the original VAR baseline reflect the combined training and inference changes and should not be attributed solely to inference-time reordering.

D Inference Latency

Table 9: VAR generation quality and end-to-end inference latency. Times are mean $\pm$ population standard deviation over $1{,}000$ images. Random and SAR share the retrained checkpoint and four-step last-scale budget.

Model	Order	Time (s/image) $\downarrow$	FID $\downarrow$	IS $\uparrow$	Prec. $\uparrow$	Rec. $\uparrow$
VAR-$d16$	1-step	0.073 ± 0.002	3.42	268.5	0.83	0.49
VAR-$d16$	random	0.093 ± 0.002	3.24	278.2	0.84	0.52
VAR-$d16$	SAR	0.093 ± 0.002	**3.05**	**286.2**	**0.86**	**0.54**
VAR-$d20$	1-step	0.088 ± 0.002	2.68	297.0	0.82	0.54
VAR-$d20$	random	0.113 ± 0.002	2.53	304.8	0.84	0.57
VAR-$d20$	SAR	0.113 ± 0.002	**2.35**	**314.5**	**0.85**	**0.59**
VAR-$d24$	1-step	0.105 ± 0.002	2.20	307.4	0.81	0.57
VAR-$d24$	random	0.135 ± 0.003	2.06	315.4	0.83	0.60
VAR-$d24$	SAR	0.135 ± 0.003	**1.88**	**325.3**	**0.84**	**0.62**
VAR-$d30$	1-step	0.133 ± 0.003	2.03	317.2	0.81	0.57
VAR-$d30$	random	0.175 ± 0.004	1.90	325.9	0.82	0.60
VAR-$d30$	SAR	0.175 ± 0.003	**1.74**	**335.5**	**0.84**	**0.62**

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Sec. 1 claim that a context-aware (SAR) generation order improves prompt alignment and reasoning-intensive instruction following on AR image models, generalizing across both patch-wise and scale-wise paradigms; Sec. 4 reports head-to-head random-vs-SAR comparisons on 8 models (MAR, MaskGen, VAR, Infinity at two scales each) on GenEval, DPG-Bench, MJHQ-30K, ImageNet FID and our long-prompt benchmark, supporting each of these claims.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The experimental discussion in Sec. 4.2 is transparent about cases where SAR does not help: MJHQ-30K FID is statistically tied with random at every scale (Table 4), and the per-category DPG-Bench L1 breakdown is mixed. The method also requires the base model to expose a per-position hidden state at each step, so it does not directly apply to single-step or strictly causal next-token AR (Sec. 3.2).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not contain formal theorems or proofs; SAR is a deterministic inference-time decoding rule (Eqs. 3 and 4) and our claims are empirical.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The SAR scoring rule and anchor definition are fully specified in Sec. 3.2; the base models, AR step counts, and anchor instantiations are listed in Sec. 4.2 and Appendix B, with the remaining sampling hyperparameters (CFG, temperature, aesthetic micro-conditioning, tokenizer/text-encoder details) given in Appendix A. SAR adds no new parameters and reuses the base models' existing conditioning embeddings as the anchor, so the experiments are reproducible from the public checkpoints listed in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The public code release covers the VAR and Infinity implementations: https://github.com/Hpesojyjx/Saccades-AR. It does not cover all experimental results. The base models and standard benchmarks are publicly available (Sec. 4 and Appendix A).

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: SAR is inference-only and adds no training, so the relevant details are sampling hyperparameters and benchmark protocols. These are stated in the Setup paragraphs of Sec. 4.2 and Appendix B, and in full in Appendix A; benchmark protocols (GenEval prompt rewriting on Infinity, DPG-Bench evaluator, MJHQ-30K subset, ImageNet ADM reference statistics, 3-seed averaging) are stated in the Evaluation paragraphs. The VAR experiments also include random-order multi-step retraining for Table 7 and a separate SAR-order retraining experiment in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All MaskGen, Infinity, and VAR results in Sec. 4 are averaged over 3 random seeds. We suppress per-cell std symbols in the tables for layout reasons but explicitly note when the random vs. SAR difference is within the multi-seed std (e.g., MJHQ-30K FID is reported as "statistically tied").

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: We do not enumerate per-experiment GPU/time figures in the main text. SAR is inference-only and adds at most one cosine similarity per uncommitted position per AR step on top of the base model's forward pass (Sec. 3.2); training costs also include the VAR multi-step retraining experiments with random ordering and SAR ordering (Appendices B and C). All other experiments use publicly released checkpoints with their original sampling settings, so reproducing them only requires the inference compute already used by the base models' authors.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We have read the NeurIPS Code of Ethics and the work conforms to it. The research uses only publicly released image generation models and benchmarks, does not involve human subjects or new data scraping.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Positive: SAR improves prompt alignment of existing image AR models without retraining and without releasing new generative weights, lowering compute cost for downstream users. Negative: like all image generation methods, the underlying base models could be misused (e.g., disinformation, deepfakes); SAR does not change the qualitative range of generations and we encourage users to follow the original models' usage guidelines and safeguards.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

Justification: We do not release new pre-trained generative models or scraped datasets. SAR is an inference-time decoding rule applied to publicly released base models (MAR, MaskGen, VAR, Infinity), each of which already has its own safeguards and usage terms.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All base models (MAR (Li et al., 2024b), MaskGen (Kim et al., 2025a), VAR (Tian et al., 2024a), Infinity (Han et al., 2025a)) and benchmarks (GenEval (Ghosh et al., 2023), DPG-Bench (Hu et al., 2024), MJHQ-30K (Li et al., 2024a), ImageNet (Deng et al., 2009)) are publicly released under their respective licenses and are cited at first use in Sec. 4 and Appendix A; we use them under those licenses without modification. The Gemini 2.5 Pro VLM judge (Gemini Team, Google, 2025) is accessed through its public API in line with its terms of service.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We introduce a complex long-prompt benchmark of 2,000 prompts together with a Gemini-2.5-Pro–judged evaluation protocol covering six reasoning categories. Sec. 4.2.1 documents the prompt construction, the evaluation protocol, and shows a representative prompt; the prompt set and judge templates will be released alongside the SAR inference code upon publication.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human subjects; all evaluation is automated (FID, GenEval, DPG-Bench mPLUG-VQA evaluator, and our Gemini-2.5-Pro VLM judge in Sec. 4.2.1).

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We use Gemini 2.5 Pro (Gemini Team, Google, 2025) as a vision–language judge in our complex long-prompt benchmark (Sec. 4.2.1): for each generated image, 10 question–answer pairs are extracted from the prompt's compositional constraints and Gemini scores the alignment between image and each question on a 1–5 scale. The judge is the only LLM in the loop; the SAR method itself uses no LLM.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.