VPG: VISUAL PREFIX GUIDANCE FOR NEXT-SCALE AUTOREGRESSIVE IMAGE AND VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive image and video generators are trained with teacher-forced histories but must sample from their own generated prefixes at inference time, creating exposure bias: early prediction errors can become part of the conditioning context and influence subsequent predictions. Existing remedies mainly modify the training or fine-tuning procedure to expose the model to perturbed or model-generated prefixes, while sampling-time guidance primarily strengthens external semantic conditions such as class labels or text prompts. We propose **Visual Prefix Guidance (VPG)**, a training-free inference-time guidance method for visual autoregressive generation. VPG contrasts predictions under the generated and corrupted prefixes and favors candidates that are both sensitive to and positively supported by the generated prefix, encouraging continuations more consistent with the realized visual history. Across class-conditional image generation with VAR, text-to-image generation with Infinity, and text-to-video generation with InfinityStar, VPG improves generation quality without retraining or fine-tuning, reducing FID on VAR by 0.40 on average and improving benchmark performance on both image and video generation.

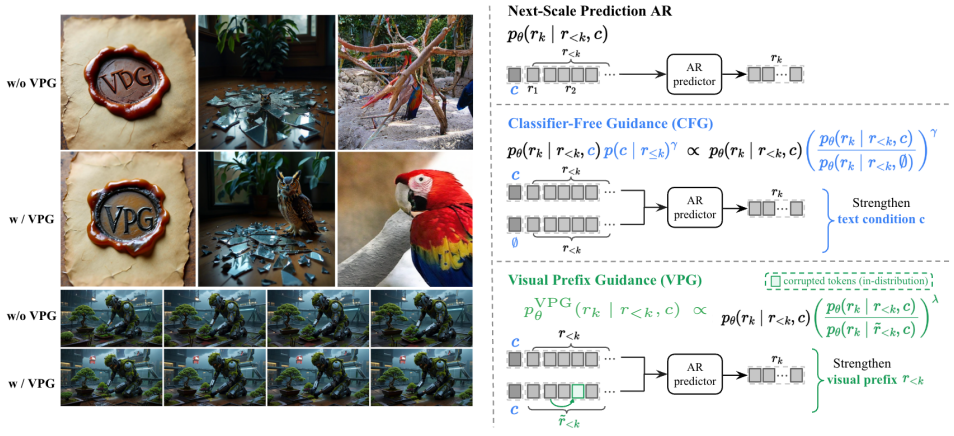


Figure 1: **Visual Prefix Guidance (VPG) strengthens dependence on the generated visual prefix**. Left: same-prompt comparisons w/o vs. w/ VPG; the first two columns use Infinity (Han et al., 2025) prompts for a "VPG" wax seal and an owl among shattered mirrors, the third uses VAR-$d30$ (Tian et al., 2024) for "macaw", and the last two rows use InfinityStar (Liu et al., 2025) for a moss-covered mech tending rooftop bonsai. Right: CFG encourages $p(c \mid r_{\le k})^\gamma$, while VPG extrapolates away from the prediction under a corrupted copy $\tilde{r}_{<k}$ of the generated prefix (Eq. 8). See Appendix F.1.

1 INTRODUCTION

Visual autoregressive models (Sun et al., 2024; Tian et al., 2024; Han et al., 2025; Liu et al., 2025) are becoming a strong alternative to diffusion models (Black Forest Labs, 2025; Kong et al., 2024; Wan et al., 2025) for both image and video generation. Instead of denoising continuous latents through an iterative reverse process, these models generate visual tokens or scales sequentially. This paradigm

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **Guidance axes in conditional visual autoregression.** In $p_\theta(r_k \mid r_{<k}, c)$, the *fixed axis* is unchanged between the two compared branches, while the *contrasted axis* is perturbed to form the guidance signal. CFG guides the external condition c, while VPG guides the generated prefix $r_{<k}$. The two contrasts can be composed to guide both axes.

Method	Fixed axis	Contrasted axis	Guided dependence
CFG	$r_{<k}$	c vs. $\emptyset$	external condition c
VPG	c	$r_{<k}$ vs. $\tilde{r}_{<k}$	generated prefix $r_{<k}$
CFG+VPG	–	both contrasts	c and $r_{<k}$

brings visual synthesis closer to GPT-style modeling (Achiam et al., 2023; Team et al., 2023), which scale well in language and multimodal systems (Team, 2024; Deng et al., 2025a).

Autoregressive models generate each new prediction from the history generated so far. We call this history the *prefix*, regardless of the prediction unit (may be a single token/token map/image frame). Teacher forcing trains these models on ground-truth prefixes, whereas inference conditions on the model's own samples, creating exposure bias (Bengio et al., 2015; Wang & Sennrich, 2020). This mismatch matters especially for next-scale prediction (Tian et al., 2024). Each step samples an entire map r_k in parallel, with sites independent given the coarser scales $r_{<k}$, and the result conditions every finer scale. A poorly supported choice is thus locked in and can propagate.

Training-time methods reduce this mismatch by optimizing on generated or perturbed histories (He et al., 2026; Bengio et al., 2015; Ren et al., 2025; Lamb et al., 2016; Liao et al., 2026), at the cost of retraining. Sampling-time guidance instead keeps the model fixed. A next-scale prediction depends on three inputs, namely the condition c, the model, and the generated prefix $r_{<k}$, and guidance contrasts it against a reference in which one input is weakened: CFG removes the condition (Ho & Salimans, 2022), and self-guidance weakens the model or its internal computation (Karras et al., 2024; Ahn et al., 2024; Hong, 2024). In both cases the generated prefix is held fixed, although it is exactly the input that exposure bias affects. Contrasting along the prefix would instead reveal, at sampling time, *how strongly each prediction is supported by the realized history.*

We propose **Visual Prefix Guidance (VPG)**, a training-free sampler that favors next-scale predictions supported by the generated prefix. Such support has a natural definition: by Bayes' rule, the ratio $p_\theta(r_k \mid r_{<k}, c)/p(r_k \mid c)$ between the model's prediction and its prefix-marginalized prediction equals the posterior-to-prior ratio of the generated prefix, so it is large when r_k is better explained by this history than by the condition alone (Sec. 4.1). Because the marginal averages over many sampled prefixes, VPG replaces it with a local reference: the prediction under a corrupted copy $\tilde{r}_{<k}$ of the current prefix, built by *same-scale full-embedding replacement*, which overwrites a random fraction of sites with embeddings of other sites at the same scale. This removes part of the prefix's evidence while keeping every input a valid embedding. At each scale, VPG extrapolates the next-scale logits away from the corrupted-prefix prediction (Fig. 6).

We evaluate VPG in class-conditional image generation with VAR (Tian et al., 2024), text-to-image generation with Infinity (Han et al., 2025), and text-to-video generation with InfinityStar (Liu et al., 2025). Across these settings, VPG improves prediction quality, all without retraining the base generator. On class-conditional VAR, VPG reduces FID by **0.40** on average across model sizes, and by as much as **0.63** for VAR-d16. On text-to-image generation, VPG improves DPG-Bench Overall from 83.03 to 83.66. On video generation, VPG improves the VBench (Huang et al., 2024) Overall score by **0.36**. Our contributions are summarized as follows:

- We introduce prefix-axis guidance, which uses the generated visual history itself as a guidance signal, and interpret it as a Bayes identity: contrasting a prediction with its prefix-marginalized counterpart favors candidates that make the generated prefix more probable.
- We propose Visual Prefix Guidance (VPG), a training-free sampler that forms a corrupted-prefix contrast via same-scale full-embedding replacement.
- VPG reduces VAR FID by 0.40 on average and raises DPG-Bench by 0.63 on Infinity, and improves VBench Overall by 0.36 on InfinityStar.

2 Related work

Visual autoregressive generation. Autoregressive (AR) models are now a competitive paradigm for image and video synthesis (Gregor et al., 2014; Parmar et al., 2018; Chen et al., 2020; Sun et al., 2024). To reduce pixel-level prediction costs, modern systems use compressed visual latents, often with VQ-VAE (Van Den Oord et al., 2017) or VQ-GAN (Esser et al., 2021) tokenizers. Advances include improved tokenization (Yu et al., 2024a; Mentzer et al., 2024; Ma et al., 2025; Li et al., 2024; Yu et al., 2024b; Ye et al., 2026), coarse-to-fine prediction in VAR (Tian et al., 2024), and bitwise tokenization with scalable generation in Infinity (Han et al., 2025; Liu et al., 2025).

Exposure bias and prefix mismatch. Exposure bias arises from training on ground-truth prefixes and sampling from generated ones (Bengio et al., 2015; Wang & Sennrich, 2020). Classical remedies either mix model-generated prefixes into training (Bengio et al., 2015; Ross et al., 2011; Zhang et al., 2019), align teacher-forced and free-running dynamics (Lamb et al., 2016), train on noisy contexts (He et al., 2026; Ren et al., 2025), or add post-training refinement objectives (Liao et al., 2026; Huang et al., 2025). Its severity has also been analyzed through error accumulation (Schmidt, 2019; Arora et al., 2022). These approaches mainly modify training or add post-training optimization, whereas our goal is to improve robustness purely at inference time.

Sampling-time guidance and self-contrasting strategies. Sampling-time guidance sharpens a fixed model's predictions by contrasting them with a weaker reference; methods differ mainly in which input the reference weakens (Tab. 1). Classifier-free guidance (CFG) removes the external condition (Ho & Salimans, 2022) and has been extended to autoregressive decoding (Xu et al., 2026). Self-guidance methods instead derive the reference from the model itself, through weaker denoisers (Karras et al., 2024), manipulated attention (Ahn et al., 2024; Hong, 2024; He et al., 2024; He & Yao, 2025), swapped features (Zhang et al., 2026), or coarser-scale priors in next-scale generation (Shin et al., 2026). These methods primarily target alignment with the condition. VPG instead targets consistency with the visual history generated so far, favoring predictions that this history specifically supports. It is thus complementary to condition guidance (Sec. 4.1).

3 Preliminaries

3.1 Next-Scale Visual Autoregressive Models

Following VAR (Tian et al., 2024) and its bitwise extension Infinity (Han et al., 2025), we model visual generation as K coarse-to-fine residual prediction stages $R = (r_1, \dots, r_K)$, ordered from the coarsest map r_1 to the finest map r_K. Each residual token map is written as $r_k \in \mathcal{V}^{h_k \times w_k}$, where $\mathcal{V}$ denotes the token space at each spatial location. In VAR, entries of r_k are VQ-VAE (Van Den Oord et al., 2017) codebook indices; in Infinity, they are represented by BSQ codes (Zhao et al., 2025). The AR model factorizes generation over these residual scales:

$$p_\theta(R \mid c) = \prod_{k=1}^{K} p_\theta(r_k \mid r_{<k}, c), \tag{1}$$

where c is an external condition, such as a class label or text prompt. The prefix $r_{<k}$ contains all previous residual scales. During training, it comes from ground-truth token maps. During inference, it is replaced by the model's sampled prefix, which for simplicity, we also denote as $r_{<k}$.

Given c and $r_{<k}$, the transformer outputs logits $\ell_k(c, r_{<k})$ for each site of the next residual map. Sampling yields $\hat{r}_k \in \mathcal{V}^{h_k \times w_k}$; the de-quantized residuals accumulate into the latent.

3.2 Classifier-Free Guidance in Visual Autoregression

The factorization in Eq. (1) exposes two conditioning inputs: the external condition c and the autoregressive prefix $r_{<k}$. We use this distinction to organize sampling-time guidance methods by which input is held fixed and which input is contrasted to form a guidance direction (Tab. 1). Sampling directly from the next-scale conditional $p_\theta(r_k \mid r_{<k}, c)$ in Eq. (1) does not necessarily lead to generations faithful to the external condition c; the sampler may favor r_k with high prefix-prior likelihood $p_\theta(r_k \mid r_{<k})$ but only weakly support c. Classifier-free guidance (CFG) (Ho & Salimans, 2022)

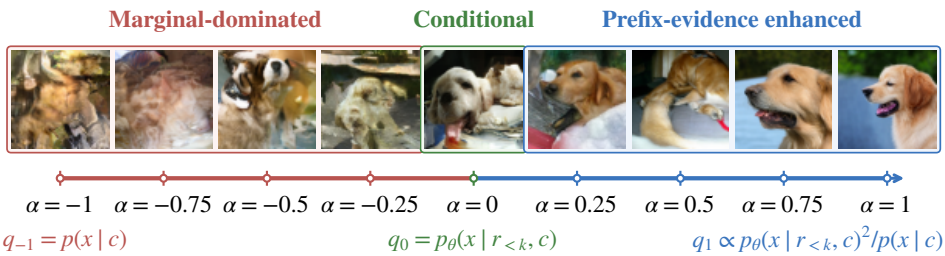


Figure 2: **Guidance along the prefix axis with a Monte Carlo marginal.** Following Eq. (6), increasing α moves sampling from the marginal ($\alpha = -1$) through the model conditional ($\alpha = 0$) to enhanced prefix-specific evidence ($\alpha > 0$). Example: golden retriever on VAR-d16, $\alpha \in \{-1, -0.75, \ldots, 1\}$, with $p(x \mid c)$ estimated from $M = 256$ Monte Carlo prefixes. All columns share the class and sampling random numbers; since guidance is applied at every scale, each strength follows its own prefix. Details in App. A.3.

addresses this at sampling time by sharpening the *posterior compatibility* of the external condition, $p(c \mid r_{\le k})$, with the prefix sequence after appending r_k, where $r_{\le k} = (r_{<k}, r_k)$. This is realized by exponentiating the compatibility term in the augmented conditional:

$$p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta(r_k \mid r_{<k}, c)\, p(c \mid r_{\le k})^\gamma, \tag{2}$$

where $\gamma \ge 0$. Bayes' rule removes the need for an explicit classifier:

$$p(c \mid r_{\le k}) \;\propto_{r_k}\; \frac{p_\theta(r_k \mid r_{<k}, c)}{p(r_k \mid r_{<k})}.$$

Therefore CFG samples from the same augmented distribution by reweighting the next-scale conditional with a conditional/unconditional likelihood ratio:

$$p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta(r_k \mid r_{<k}, c) \left(\frac{p_\theta(r_k \mid r_{<k}, c)}{p(r_k \mid r_{<k})} \right)^\gamma,$$

up to normalization over r_k (App. A.1). Approximating the unconditional predictive with the learned null-condition branch, $p(r_k \mid r_{<k}) \approx p_\theta(r_k \mid r_{<k}, \emptyset)$, gives the logit rule

$$\ell_k^{\mathrm{CFG}} \;=\; \ell_k^c + \gamma\big(\ell_k^c - \ell_k^\emptyset\big) \;=\; (1+\gamma)\,\ell_k^c - \gamma\,\ell_k^\emptyset. \tag{3}$$

Here $\ell_k^c = \ell_k(c, r_{<k})$ and $\ell_k^\emptyset = \ell_k(\emptyset, r_{<k})$ share the same prefix. Increasing γ favors candidates with higher condition compatibility. CFG is thus condition-axis guidance at a fixed generated prefix.

4 Method

4.1 Visual Prefix Guidance as Log-Ratio Guidance

Sec. 3.2 casts CFG as log-ratio guidance along the condition axis. VPG applies the same construction along the prefix axis: fixing the condition c and a step k, it reweights $p_\theta(r_k \mid r_{<k}, c)$ by how strongly each candidate supports the generated prefix $r_{<k}$ rather than the condition alone.

To measure this support, we compare the model's prediction under $r_{<k}$ with its prediction averaged over prefixes from a reference distribution $p(r_{<k} \mid c)$:

$$p(r_k \mid c) = \sum_{r_{<k}} p(r_{<k} \mid c)\, p_\theta(r_k \mid r_{<k}, c). \tag{4}$$

Bayes' rule under the joint $p(r_{<k} \mid c)\, p_\theta(r_k \mid r_{<k}, c)$ gives

$$\frac{p_\theta(r_k \mid r_{<k}, c)}{p(r_k \mid c)} = \frac{p(r_{<k} \mid r_k, c)}{p(r_{<k} \mid c)}, \tag{5}$$

so the ratio is large when r_k is evidence for this particular history, just as the CFG ratio, proportional in r_k to $p(c \mid r_{\le k})$, is large when r_k is evidence for the condition. The identity holds equally at each

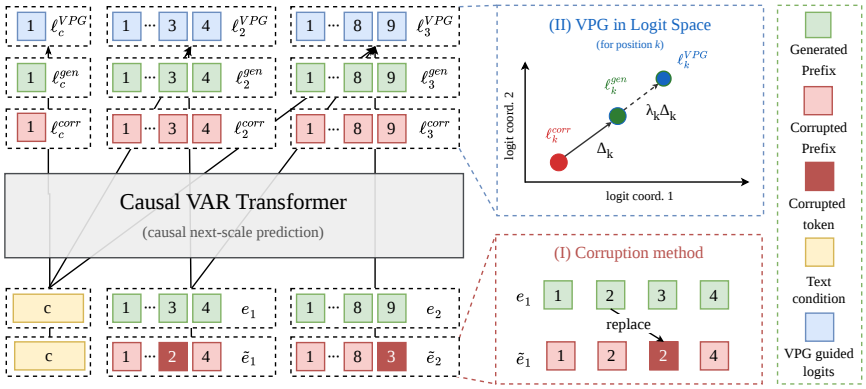


Figure 3: **Overview of Visual Prefix Guidance.** VPG contrasts logits from the generated prefix $r_{<k}$ and a same-scale corrupted prefix $\tilde{r}_{<k}$ in one frozen transformer, then extrapolates before sampling the next token map. The corrupted branch is a local counterfactual reference.

site's prediction head, for a candidate token x with marginal $p(x \mid c)$. Sharpening each head by this ratio gives the guidance family

$$\begin{aligned} q_\alpha(x \mid r_{<k}, c) &\propto p_\theta(x \mid r_{<k}, c) \left[\frac{p_\theta(x \mid r_{<k}, c)}{p(x \mid c)}\right]^\alpha \\ &= p(x \mid c) \left[\frac{p(r_{<k} \mid x, c)}{p(r_{<k} \mid c)}\right]^{1+\alpha}, \qquad \alpha \geq -1, \end{aligned} \tag{6}$$

which samples the marginal at $\alpha = -1$, recovers the model at $\alpha = 0$, and increasingly favors prefix-supported candidates for $\alpha > 0$.

The marginal has no closed form. An ancestral Monte Carlo estimate confirms this behavior (Fig. 2; App. A.3), but every bank entry is a full rollout, and a large vocabulary needs many of them before the estimate is stable. VPG therefore keeps the form of Eq. (6) and replaces the marginal with a reference computed from the current prefix.

A local counterfactual reference. VPG obtains its reference prediction by corrupting the current generated prefix (Fig. 3). Whereas the marginal averages over other prefixes and requires many rollouts, this reference keeps the current prefix and perturbs only its visual configuration. For the generated prefix $r_{<k}$ and corruption randomness ξ, let $\tilde{r}_{<k} = C(r_{<k}; \xi)$ and define

$$p_{\text{ref}}(r_k \mid c) := p_\theta(r_k \mid \tilde{r}_{<k}, c), \tag{7}$$

where $C(r_{<k}; \xi)$ removes the evidence of a random fraction of the prefix's sites while retaining its scale structure and external condition. The reference p_{ref} implicitly depends on $r_{<k}$ and ξ, which are held fixed below. The VPG distribution is defined as

$$p_\theta^{\text{VPG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta(r_k \mid r_{<k}, c) \left(\frac{p_\theta(r_k \mid r_{<k}, c)}{p_{\text{ref}}(r_k \mid c)}\right)^\lambda, \qquad \lambda \geq 0. \tag{8}$$

VPG thus favors candidates whose probability drops when the prefix is corrupted, i.e., candidates with greater support from the generated prefix. Eq. (8) has the form of Eq. (6) with p_{ref} in place of the marginal: Eq. (5) motivates the direction of the contrast, and the choice of reference sets which evidence is measured. For any two candidates a and b,

$$\frac{p_\theta^{\text{VPG}}(a \mid r_{<k}, c)}{p_\theta^{\text{VPG}}(b \mid r_{<k}, c)} = \frac{p_\theta(a \mid r_{<k}, c)}{p_\theta(b \mid r_{<k}, c)} \left[\frac{p_\theta(a \mid r_{<k}, c)/p_{\text{ref}}(a \mid c)}{p_\theta(b \mid r_{<k}, c)/p_{\text{ref}}(b \mid c)}\right]^\lambda. \tag{9}$$

Hence, increasing λ monotonically raises the relative sampling odds of candidates with larger clean-to-corrupted prefix contrast.

Taking logarithms of Eq. (8) gives the logit rule:

$$\ell_k^{\text{VPG}} \;=\; \ell_k^{\text{gen}} + \lambda\big(\ell_k^{\text{gen}} - \ell_k^{\text{corr}}\big) \;=\; (1+\lambda)\,\ell_k^{\text{gen}} - \lambda\,\ell_k^{\text{corr}}, \tag{10}$$

Table 2: **ImageNet class-conditional generation with VAR.** FID is measured with 50,000 samples using the ADM evaluation suite. d16–d30 use 256×256 generation, while d36 uses 512×512. CFG and VPG entries are means over five seeds (App. C.1); SSG values are as reported by Shin et al. (2026). Best values within each depth are bolded.

Model	#Params	Resolution	CFG	SSG	VPG (Ours)
VAR-d16	310M	256^2	3.37	3.27	**2.74**
VAR-d20	600M	256^2	2.71	2.49	**2.26**
VAR-d24	1.0B	256^2	2.16	2.20	**1.84**
VAR-d30	2.0B	256^2	1.97	**1.68**	1.84
VAR-d36	2.3B	512^2	2.65	2.39	**2.17**

where $\ell_k^{\text{gen}} \equiv \ell_k(c, r_{<k})$ and $\ell_k^{\text{corr}} \equiv \ell_k(c, \tilde{r}_{<k})$ are the scale-k logits under the generated and corrupted prefixes. The rule is applied site-wise before sampling, to the parallel heads in VAR-style models and to the bitwise logits in Infinity-style models, and adds no auxiliary model or retraining.

Composition with CFG. CFG and VPG are composed directly in logit space. We first apply CFG separately to the genuine-prefix and corrupted-prefix branches,

$$g_k^{\text{gen}} = \ell_k(c, r_{<k}) + \underbrace{\gamma\big(\ell_k(c, r_{<k}) - \ell_k(\emptyset, r_{<k})\big)}_{\text{CFG on genuine prefix}}, \tag{11}$$

$$g_k^{\text{corr}} = \ell_k(c, \tilde{r}_{<k}) + \underbrace{\gamma\big(\ell_k(c, \tilde{r}_{<k}) - \ell_k(\emptyset, \tilde{r}_{<k})\big)}_{\text{CFG on corrupted prefix}}. \tag{12}$$

We then apply VPG to these CFG-guided branch logits:

$$\ell_k^{\text{CFG+VPG}} = \underbrace{g_k^{\text{gen}}}_{\text{CFG-guided base}} + \underbrace{\lambda\big(g_k^{\text{gen}} - g_k^{\text{corr}}\big)}_{\text{VPG}}. \tag{13}$$

This composes condition- and prefix-axis guidance; the likelihood derivation is in App. A.4.

4.2 Constructing a Corrupted Prefix

We instantiate the corruption C of Eq. (7) by *same-scale full embedding replacement*: across the prefix, a uniformly random fraction n_p of sites have their full embedding (visual code with its scale-position encoding) replaced with that of another site at the same scale, where n_p is the *corruption fraction* (See ablations in Sec. 5.4). Because donor embeddings come from the same generated prefix and the same scale, the branch preserves the model, external condition, scale geometry, and embedding format while removing the selected sites' own evidence from the prefix. Constructing a weaker reference branch from the model's own activations to avoid additional post-training is a recurring pattern in inference-time guidance (Karras et al., 2024; Ahn et al., 2024; Hong, 2024; Xu et al., 2026). Sec. 5.4 ablates each component of this construction.

Formally, each prefix segment $j \in \{1, \dots, k-1\}$ uses the downsampled cumulative feature at the corresponding scale. It is embedded with the model's visual projection and scale-position encoding.

$$e_{j,u} = \text{Emb}(\bar{F}_j[u]) + \text{PosEmb}(j, u), \qquad 1 \le j < k, \tag{14}$$

where u indexes spatial sites. When segment j enters the corrupted branch, each of its sites is selected independently with probability n_p and receives a donor $\pi(j, u) = (j, u')$ drawn uniformly, with replacement, from the same scale. The segment is corrupted once and reused through the key–value cache; $\mathcal{S}_k$ collects the sites selected in $r_{<k}$.

$$\tilde{e}_{j,u} = \begin{cases} e_{j,u'}, & (j,u) \in \mathcal{S}_k, \quad (j,u') = \pi(j,u), \\ e_{j,u}, & (j,u) \notin \mathcal{S}_k. \end{cases} \tag{15}$$

The corrupted-prefix prediction $p_\theta(r_k \mid \tilde{r}_{<k}, c)$ uses the replaced embeddings from Eq. (15).



Figure 4: **DPG-Bench examples on Infinity-2B**. CFG renders the main subject but drops secondary elements of long prompts (**highlighted**); VPG recovers them, including object-in-object relations (sneakers in the tub, a candle in the spout, apples in bags) and spatial relations (an owl on a branch overhead). Prompts are abbreviated.

5 Experiments

Our experiments cover class-conditional image generation with VAR (Tian et al., 2024), text-to-image generation with Infinity (Han et al., 2025), and text-to-video generation with InfinityStar (Liu et al., 2025). We first report main results in these three settings, then ablate the corrupted-prefix construction and VPG hyperparameters on VAR.

5.1 Experimental setup

We evaluate released checkpoints without retraining or architectural changes. *Every baseline uses the sampling configuration that its authors tuned and released for the benchmark we evaluate*, so VPG is always compared with each model's official CFG setting rather than with a weakened baseline. VPG is added on top of these settings, none of which is changed (App. B.1). For class-conditional image generation, we use official VAR checkpoints at depths 16/20/24/30 at 256×256 and VAR-d36 at 512×512, following the standard ImageNet (Deng et al., 2009) class-conditional protocol with 50,000 generated samples. We use the ADM evaluation suite (Dhariwal & Nichol, 2021) and report Fréchet Inception Distance (FID) (Heusel et al., 2017) as the main metric. For text-to-image generation, we evaluate the released Infinity checkpoint on DPG-Bench (Hu et al., 2024). For text-to-video generation, we evaluate the InfinityStar checkpoint on all 946 VBench prompts (Huang et al., 2024).

5.2 Main results

Class-conditional generation. Tab. 2 compares VPG at its best-FID operating point with CFG and the scale-wise self-guidance baseline SSG (Shin et al., 2026). VPG improves on the official CFG setting at every depth, by 0.40 FID on average and by up to 0.63 on d16 (3.37→2.74), and the gain carries over to 512×512 on d36 (2.65→2.17). The improvement is robust: VPG wins for every seed at every depth (App. C.1), and we also report the four held-out seeds alone, which still give 0.39 (App. C.1). VPG outperforms SSG on four of the five depths, with SSG ahead on d30. The coarse-scale schedule retains the gain at 1.22× the cost (App. C.2, Tab. 12, Fig. 7).

Table 4: **Text-to-video evaluation on VBench. bold** and underlined mark best and second-best results; **green** values report changes relative to the official InfinityStar sampler. VPG gives InfinityStar the best Scene, Multi-object, and Semantic scores while improving Overall.

Model	#Params	Human Action	Scene	Multi object	Appearance Style	Quality Score	Semantic Score	Overall
Diffusion Models								
AnimateDiff-V2	1.5B	92.60	50.19	36.88	22.42	82.90	69.75	80.27
VideoCrafter-2.0	1.5B	95.00	55.29	40.66	**25.13**	82.20	73.42	80.44
OpenSora V1.2	1.1B	85.80	42.47	58.41	23.89	80.71	73.30	79.23
Show-1	6B	95.60	47.03	45.47	23.06	80.42	72.98	78.93
Gen-3	–	96.40	54.57	53.64	24.31	84.11	75.17	82.32
CogVideoX-5B	5B	**99.40**	53.20	62.11	24.91	82.75	77.04	81.61
HunyuanVideo	13B	94.40	53.88	68.55	19.80	85.09	75.82	83.24
Goku	2B	97.60	57.08	79.48	23.08	85.60	81.87	**84.85**
Wan 2.1	14B	98.80	53.67	81.44	21.13	**85.64**	80.95	84.70
Autoregressive Models								
Nova	0.6B	95.20	54.06	77.52	20.92	80.39	79.05	80.12
Emu3	8B	77.71	37.11	44.64	20.92	84.09	68.43	80.96
InfinityStar	8B	98.00	54.51	87.50	20.54	84.14	82.74	83.86
+ VPG (Ours)	8B	99.00 (+1.00)	**57.19** (+2.68)	**88.57** (+1.07)	20.75 (+0.21)	84.36 (+0.22)	**83.65** (+0.91)	84.22 (+0.36)

Text-to-image generation. Tab. 3 compares the official Infinity-2B sampler with and without VPG (n_p=0.05, λ=0.8) on DPG-Bench, whose long prompts describe several entities together with attributes and relations. Other rows are reported by their authors (Podell et al., 2024; Chen et al., 2024b; Esser et al., 2024; Wang et al., 2024). VPG raises the Overall score from 83.03 to 83.66, and the gain is positive for every seed (App. C.1). Fig. 4 shows the typical effect: elements that CFG drops from a long prompt, such as a secondary object or an object-in-object relation, are rendered with VPG.

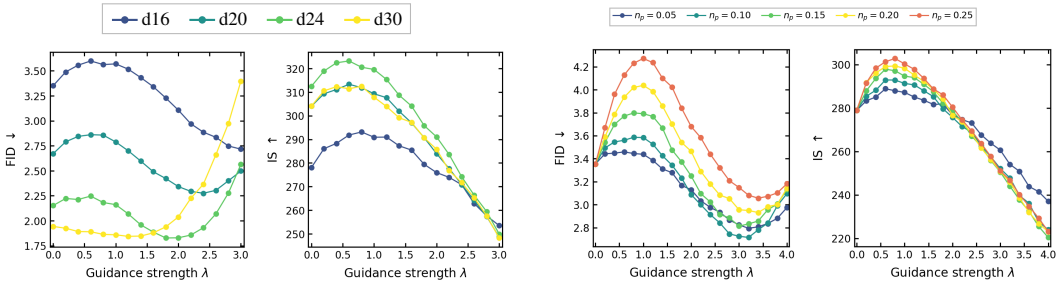
Table 3: **DPG-Bench evaluation with Infinity.** VPG is applied to the official Infinity-2B sampler; our results are averaged over five seeds (App. 10).

Method	#Params	Overall↑
SDXL	2.6B	74.65
PixArt-Σ	0.6B	80.54
SD3 (d=24)	2B	84.08
Emu3	8.5B	81.60
Infinity	2B	83.03
+ VPG	2B	83.66 (+0.63)

Text-to-video generation. Tab. 4 compares the released InfinityStar checkpoint with and without VPG. Other rows are reported by their authors (Guo et al., 2024; Chen et al., 2024a; Zheng et al., 2024; Zhang et al., 2025; Runway, 2024; Yang et al., 2025; Kong et al., 2024; Chen et al., 2025; Wan et al., 2025; Deng et al., 2025b; Wang et al., 2024). We apply same-scale full-embedding replacement with n_p=0.05 and λ=0.5. A smaller corruption fraction is important for video generation, as long spatiotemporal rollouts are more sensitive to prefix perturbations; see Appendix C.2 for more details. VPG improves InfinityStar's Overall score from 83.86 to 84.22. The most notable gains are semantic: VPG achieves the best Scene score of 57.19, the best Multi-object score of 88.57, and the best Semantic score of 83.65 among all listed models.

5.3 Qualitative Study

Across the three settings, VPG mainly improves structural coherence and the retention of prompted details. On class-conditional VAR, it gives more coherent objects and more stable layouts than CFG (Fig. 12), for example turning an occluded, malformed macaw into a complete bird (Fig. 1). On DPG-Bench, VPG more reliably renders the details that CFG drops: objects placed in or on others (a candle in a bottle's spout, binoculars on a window ledge), counts and attributes (ten triangles and five white squares), and short text ("deep learning" spelled in carrot slices) (Figs. 4, 13–14). On InfinityStar, VPG keeps the prompted concept and scene layout more stable across frames, preserving, for example, the folded-paper surface of an origami whale and the vegetation of a subway greenhouse (Fig. 15). Full prompts and settings are in App. F.



(Left) FID, IS vs. λ (n_p=0.1, VAR-d16 to d30). **(Right)** FID, IS vs. λ for corruption levels.

Figure 5: **VPG sweeps on class-conditional VAR (ImageNet 256×256). Left:** guidance strength λ with fixed corruption fraction n_p=0.1, across four VAR capacities. **Right:** the same λ sweep on VAR-d16 with the corruption fraction varied over $n_p \in \{0.05, 0.10, 0.15, 0.20, 0.25\}$; larger n_p amplifies the FID/IS response to λ, and smaller n_p gives a flatter response.

5.4 Ablations

We ablate VPG on VAR under the ImageNet 256×256 ADM protocol of Tab. 2, asking (i) what kind of corrupted prefix makes a useful weak reference and (ii) how sensitive VPG is to λ and n_p.

Effect of corrupted-prefix construction. Tab. 5 ablates how the corrupted prefix $\tilde{r}_{<k}$ is built on VAR-d16 with fixed n_p=0.1. The replacement rules are formalized in Appendix D.1.

Table 5: **Corrupted-prefix variants on VAR-d16** (n_p=0.1): best FID over $\lambda \in [0.2, 4]$, all runs at seed 42 (CFG-only: 3.40; the corresponding five-seed means in Tab. 2 are 3.37 and 2.74).

Variant	Best FID↓ ($\lambda^\star$)
None (CFG only)	3.40
Random codebook	3.52 (0.2)
Same-scale token	3.44 (0.2)
Same-scale position	3.55 (0.2)
Same-scale embed. (Ours)	**2.72** (3.2) **(-0.68)**

The three incomplete or off-manifold corruptions never improve on CFG: their best FIDs (3.52, 3.44 and 3.55) are all reached at the smallest tested λ, that is, with the guidance almost switched off, and at the VPG operating point λ=3.2 they degrade to 15.63, 13.54 and 7.15. Only same-scale full-embedding replacement improves over CFG, to 2.72 at λ=3.2. This supports the design criterion in Sec. 4.2: corruption should act as a controlled contrast rather than generic degradation. Simply weakening the prefix is insufficient; the reference should remove prefix evidence while keeping every input a valid same-scale code–position embedding. The corresponding FID/IS curves are provided in Appendix D.1. Weak branches that are not a corrupted prefix at all, such as dropped sites, attention perturbation, entropy-matched heating or a random direction of the same norm, do not improve on CFG either (App. D.2).

Sensitivity to guidance strength and corruption fraction. With n_p=0.1 fixed (Fig. 5, left), VPG reaches a lower FID than the CFG-only baseline at every VAR capacity, and the best λ decreases as model size grows. On VAR-d16 (right), a larger n_p amplifies the response to λ. Moderate corruption with a sufficiently large λ gives the best FID, matching the operating point in Tab. 2. VPG also improves coverage: it raises recall at every model size (App. C.1) through more varied global layouts, which VAR fixes in the first four of its ten scales, while each image stays consistent with its own layout (App. D.4). It also reaches a better FID–recall trade-off than simply adding randomness: at every matched recall, VPG attains a lower FID than a lower CFG scale or a higher sampling temperature (Fig. 11).

6 Conclusion

Visual Prefix Guidance (VPG) makes the generated visual history an explicit guidance signal for autoregressive generation. Where CFG contrasts the external condition, VPG contrasts predictions under the generated prefix and a corrupted copy of it, favoring candidates with greater support from the generated prefix; it requires no retraining or auxiliary model and composes directly with CFG. Across VAR, Infinity, and InfinityStar, VPG reduces VAR FID by 0.40 on average and improves DPG-Bench and VBench.

Reproducibility statement

Experimental protocols, evaluation settings, and VPG hyperparameter choices are documented in Sec. 5 and App. B, including the use of released checkpoints and public benchmark protocols. Full theoretical derivations and additional supporting analyses are provided in the appendix. These details are intended to make the reported inference-time procedure and evaluations reproducible without retraining the underlying generators.

AI use statement

Generative AI tools were used to aid and polish writing, to support retrieval and discovery of related work, and to assist with research ideation and execution. The authors reviewed and verified AI-assisted outputs against the relevant derivations, experimental results, code, and cited sources, and take responsibility for the final content of the paper.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *European Conference on Computer Vision*, pp. 1–17. Springer, 2024.

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Cheung. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 700–710, 2022. doi: 10.18653/v1/2022.findings-acl.58. URL https://aclanthology.org/2022.findings-acl.58/.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in Neural Information Processing Systems*, 28, 2015.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. VideoCrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7310–7320, 2024a.

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-Σ: Weak-to-strong training of diffusion transformer for 4K text-to-image generation. In *European Conference on Computer Vision*, 2024b.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pp. 1691–1703. PMLR, 2020.

Shoufa Chen, Chongjian Ge, Yuqi Zhang, Yida Zhang, Fengda Zhu, Hao Yang, Hongxiang Hao, Hui Wu, Zhichao Lai, Yifei Hu, Ting-Che Lin, Shilong Zhang, Fu Li, Chuan Li, Xing Wang, Yanghua Peng, Peize Sun, Ping Luo, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Goku: Flow based video generative foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025a.

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. In *International Conference on Learning Representations*, 2025b.

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*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633, 2024.

Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *International Conference on Machine Learning*, pp. 1242–1250. PMLR, 2014.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. AnimateDiff: Animate your personalized text-to-image diffusion models without specific tuning. In *International Conference on Learning Representations*, 2024.

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 (CVPR)*, pp. 15733–15744, June 2025.

Qiyuan He and Angela Yao. Conceptrol: Concept control of zero-shot personalized image generation. *arXiv preprint arXiv:2503.06568*, 2025.

Qiyuan He, Jinghao Wang, Ziwei Liu, and Angela Yao. AID: Attention interpolation of text-to-image diffusion. In *Advances in Neural Information Processing Systems*, 2024.

Qiyuan He, Yicong Li, Haotian Ye, Jinghao Wang, Xinyao Liao, Pheng-Ann Heng, Stefano Ermon, James Zou, and Angela Yao. reAR: Rethinking visual autoregressive models via token-wise consistency regularization. In *International Conference on Learning Representations*, 2026.

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. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention. *Advances in Neural Information Processing Systems*, 37:66743–66772, 2024.

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.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, 2025.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models, 2024. URL https://arxiv.org/abs/2412.03603.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Alex M. Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37: 56424–56445, 2024.

Xinyao Liao, Qiyuan He, Kai Xu, Xiaoye Qu, Yicong Li, Wei Wei, and Angela Yao. VA-π: Variational policy alignment for pixel-aware autoregressive generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Jinlai Liu, Jian Han, Bin Yan, Hui Wu, Fengda Zhu, Xing Wang, Yi Jiang, Bingyue Peng, and Zehuan Yuan. InfinityStar: Unified spacetime autoregressive modeling for visual generation. In *Advances in Neural Information Processing Systems*, 2025.

Chuofan Ma, Yi Jiang, Junfeng Wu, Jihan Yang, Xin Yu, Zehuan Yuan, Bingyue Peng, and Xiaojuan Qi. UniTok: A unified tokenizer for visual generation and understanding. In *Advances in Neural Information Processing Systems*, 2025.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *International Conference on Learning Representations*, 2024.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International conference on machine learning*, pp. 4055–4064. PMLR, 2018.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, 2024.

Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Beyond next-token: Next-X prediction for autoregressive visual generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Runway. Runway Gen-3 Alpha: AI video generation with temporal control. https://runway.com/research/introducing-gen-3-alpha, 2024. Blog post, June 17, 2024.

Seyedmorteza Sadat, Otmar Hilliges, and Romann M. Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models. In *International Conference on Learning Representations*, 2025.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Florian Schmidt. Generalization in generation: A closer look at exposure bias. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pp. 157–167, 2019. doi: 10.18653/v1/D19-5616. URL https://aclanthology.org/D19-5616/.

Youngwoo Shin, Jiwan Hur, and Junmo Kim. SSG: Scaled spatial guidance for multi-scale visual autoregressive generation. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=S6oLw7VixT.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Aaron Van Den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Chaojun Wang and Rico Sennrich. On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Dongli Xu, Aleksei Tiulpin, and Matthew B. Blaschko. SoftCFG: Uncertainty-guided stable guidance for visual autoregressive model. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=G7tqQ5Upcs.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, 2025.

Haotian Ye, Qiyuan He, Jiaqi Han, Puheng Li, Jiaojiao Fan, Zekun Hao, Fitsum Reda, Yogesh Balaji, Huayu Chen, Sheng Liu, et al. InfoTok: Adaptive discrete video tokenizer via information-theoretic compression. In *International Conference on Learning Representations*, 2026.

Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, et al. Language model beats diffusion – tokenizer is key to visual generation. In *International Conference on Learning Representations*, 2024a.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024b.

David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. Show-1: Marrying pixel and latent diffusion models for text-to-video generation. *International Journal of Computer Vision*, 133(4):1879–1893, 2025.

Weijia Zhang, Yuehao Liu, Shanyan Guan, Wu Ran, Yanhao Ge, Wei Li, and Chao Ma. Guiding a diffusion model by swapping its tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4334–4343, 2019. doi: 10.18653/v1/P19-1426. URL https://aclanthology.org/P19-1426/.

Yue Zhao, Yuanjun Xiong, and Philipp Krähenbühl. Image and video tokenization with binary spherical quantization. In *International Conference on Learning Representations*, 2025.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-Sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

APPENDIX OVERVIEW

The appendix follows the order of the main text. Each part lists the claims it supports.

Part	Content	Supports
A	Derivations: CFG and prefix-axis guidance, the Monte Carlo marginal, composition with CFG and APG	Secs. 3.2, 4.1
B	Implementation: models, samplers, hyperparameters, baseline strength (CFG sweep); assets, licenses and compute	Sec. 5.1
C	Additional results: multi-seed statistics on VAR and DPG-Bench; compute cost and the coarse-scale schedule on VAR and InfinityStar	Sec. 5.2
D	Analysis of the corrupted-prefix reference: replacement rules and their distance to the marginal, other weak branches, per-scale statistics; precision, recall and diversity (FID–recall comparison with CFG scale and temperature, prefix branching)	Sec. 5.4
E	Limitations and broader impacts	–
F	Qualitative results: teaser details; same-seed VAR-d36, DPG-Bench and InfinityStar comparisons, with a cross-model summary	Sec. 5.3

A DERIVATIONS

CFG (Sec. 3.2) and VPG (Sec. 4.1) are both log-ratio rules. Each starts from a posterior term that the sampler should favor and uses Bayes' rule to write it as a ratio of two predictions; the two rules differ in the input that the ratio contrasts. We derive them in parallel (App. A.1–A.2), describe the Monte Carlo experiment of Fig. 2 (App. A.3), and derive the composed sampler (App. A.4).

A.1 CFG: FAVORING EVIDENCE FOR THE CONDITION

Augmented conditional. CFG exponentiates the posterior compatibility $p(c \mid r_{\le k})$ of the condition with the extended prefix $r_{\le k} = (r_{<k}, r_k)$:

$$p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta(r_k \mid r_{<k}, c)\, p(c \mid r_{\le k})^\gamma, \qquad \gamma \ge 0. \tag{16}$$

Bayes decomposition along the AR chain. Bayes' rule applied to $p(r_k, c \mid r_{<k})$ gives

$$p_\theta(r_k \mid r_{<k}, c) \;=\; \frac{p(c \mid r_{\le k})\, p_\theta(r_k \mid r_{<k})}{p(c \mid r_{<k})}. \tag{17}$$

Since $p(c \mid r_{<k})$ does not depend on r_k,

$$\log p(c \mid r_{\le k}) \;=\; \log p_\theta(r_k \mid r_{<k}, c) - \log p_\theta(r_k \mid r_{<k}) + \mathrm{const}, \tag{18}$$

the autoregressive analogue of the implicit-classifier identity behind diffusion CFG. Substituting Eq. (18) into Eq. (16) gives the ratio form of Sec. 3.2,

$$p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta(r_k \mid r_{<k}, c) \left(\frac{p_\theta(r_k \mid r_{<k}, c)}{p(r_k \mid r_{<k})} \right)^\gamma, \tag{19}$$

so the condition posterior is never evaluated by a separate classifier.

From two predictions to guided logits. With the learned null condition, $p(r_k \mid r_{<k}) \approx p_\theta(r_k \mid r_{<k}, \emptyset)$, the logarithm of Eq. (19) is

$$\log p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \;=\; (1+\gamma)\, \log p_\theta(r_k \mid r_{<k}, c) - \gamma\, \log p_\theta(r_k \mid r_{<k}, \emptyset) + \mathrm{const}, \tag{20}$$

which, applied to the pre-softmax logits, is the rule $\ell_k^{\mathrm{CFG}} = (1+\gamma)\, \ell_k^c - \gamma\, \ell_k^\emptyset$ of Eq. (3). The common parameterization $\ell_k^\emptyset + s\,(\ell_k^c - \ell_k^\emptyset)$ corresponds to $s = 1 + \gamma$.

A.2 Prefix-axis guidance and the VPG reference

Bayes identity. Fix c and a step k, and let $p(r_{<k} \mid c)$ be a reference distribution over prefixes, such as the model's own ancestral prefix distribution. Together with the frozen model it defines the joint $p(r_{<k} \mid c)\, p_\theta(r_k \mid r_{<k}, c)$, whose marginal over prefixes is $p(r_k \mid c)$ (Eq. (4)) and whose posterior over prefixes is

$$p(r_{<k} \mid r_k, c) = \frac{p(r_{<k} \mid c)\, p_\theta(r_k \mid r_{<k}, c)}{p(r_k \mid c)}.$$

Dividing both sides by $p(r_{<k} \mid c)$ gives Eq. (5). The posterior here is over prefixes, not over the condition: the ratio is large when observing r_k makes the realized prefix more probable than it is a priori, just as the CFG ratio is large when r_k makes the condition more probable.

Signed family. The identity holds at every site's prediction head. For a candidate token x with head prediction $p_\theta(x \mid r_{<k}, c)$ and marginal $p(x \mid c)$, Eq. (6) can be written as

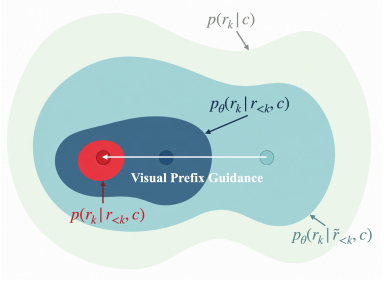
$$q_\alpha(x \mid r_{<k}, c) \;\propto\; \frac{p_\theta(x \mid r_{<k}, c)^{1+\alpha}}{p(x \mid c)^{\alpha}} \;=\; p(x \mid c) \left(\frac{p(r_{<k} \mid x, c)}{p(r_{<k} \mid c)} \right)^{1+\alpha}, \qquad \alpha \geq -1. \tag{21}$$

In log space, $\log q_\alpha = (1+\alpha) \log p_\theta - \alpha \log p + \text{const}$. For $-1 \leq \alpha \leq 0$ this is a geometric interpolation between the marginal and the model; for $\alpha > 0$ it extrapolates away from the marginal, the same logit extrapolation as CFG with the marginal in place of the null-condition prediction. A geometric interpolation differs from the arithmetic mixture $(1-\beta)\, p(x \mid c) + \beta\, p_\theta(x \mid r_{<k}, c)$, even though both share the endpoints.

The deployed reference. VPG keeps this form and replaces the marginal with the corrupted-prefix prediction $p_{\text{ref}}(x \mid c) = p_\theta(x \mid \tilde{r}_{<k}, c)$ of Eq. (7):

$$q_\lambda^{\text{VPG}}(x \mid r_{<k}, c) \;\propto\; \frac{p_\theta(x \mid r_{<k}, c)^{1+\lambda}}{p_{\text{ref}}(x \mid c)^{\lambda}}, \qquad \lambda \geq 0, \tag{22}$$

whose logarithm is the logit rule of Eq. (10). The two references answer different questions (Tab. 6). The marginal compares the realized prefix with other prefixes the model could have generated; the corrupted prefix compares it with a copy of itself in which part of the evidence is removed, and costs one extra forward branch instead of many rollouts. Eq. (22) therefore coincides with q_α only if $p_{\text{ref}} = p(x \mid c)$, which we do not assume: the Bayes identity fixes the direction of the contrast, and the choice of reference sets which evidence it measures. App. D studies which references work. Fig. 6 illustrates the construction.



	Marginal (Eq. 21)	VPG (Eq. 22)
Reference	$p(x \mid c)$	$p_\theta(x \mid \tilde{r}_{<k}, c)$
Compares $r_{<k}$ with	other prefixes	a corrupted copy
Exponent	$\alpha \geq -1$	$\lambda \geq 0$
Extra cost	M rollouts per class	one branch per step

Table 6: **Two references for prefix-axis guidance.**

Figure 6: **Schematic of VPG at one prediction step.** Nested regions sketch the prefix marginal $p(r_k \mid c)$ (outer), the corrupted-prefix prediction $p_\theta(r_k \mid \tilde{r}_{<k}, c)$ and the model prediction $p_\theta(r_k \mid r_{<k}, c)$. VPG extrapolates from the corrupted-prefix prediction through the model prediction (arrow) and concentrates on candidates supported by the generated prefix (red: the distribution of Eq. (8)).

A.3 Monte Carlo estimate of the prefix marginal

Fig. 2 samples from Eq. (6) with an estimated marginal. It checks the direction of the guidance family and is not the VPG sampler. For one class of VAR-d16 (golden retriever), we draw a bank of $M = 256$ prefixes $r_{<k}^{(1)}, \ldots, r_{<k}^{(M)} \sim p(\cdot \mid c)$ by ancestral sampling without CFG, store the head

distributions of every bank rollout at every scale, and keep the bank fixed. At scale k and site u, the marginal head is the bank average

$$\widehat{p}_{M,u}(x \mid c) = \frac{1}{M} \sum_{i=1}^{M} p_{\theta,u}(x \mid r_{<k}^{(i)}, c). \tag{23}$$

This is a per-site estimate, matching the per-site sampling of a next-scale head; it is not the marginal of the whole map. Each column of Fig. 2 applies Eq. (6) at every scale with this estimate and the same class and sampling random numbers. The estimate needs M full rollouts per class before sampling can start, which is the cost that the local reference of VPG avoids (Tab. 6). App. D.1 uses the same estimator to compare the corrupted-prefix rules with the marginal.

A.4 Composition with CFG and APG

The sampler composes the two guidance operations sequentially. First, CFG is applied to each prefix branch $a_{<k} \in \{r_{<k}, \tilde{r}_{<k}\}$, which by App. A.1 gives the branch logits $g_k(a_{<k})$:

$$\begin{aligned} p_\theta^{\mathrm{CFG}}(r_k \mid a_{<k}, c) \;&\propto\; p_\theta(r_k \mid a_{<k}, c)\, p(c \mid a_{<k}, r_k)^\gamma, \\ g_k(a_{<k}) \;&=\; (1+\gamma)\, \ell_k(c, a_{<k}) - \gamma\, \ell_k(\emptyset, a_{<k}). \end{aligned} \tag{24}$$

Second, VPG contrasts the two CFG-guided branches:

$$p_\theta^{\mathrm{CFG+VPG}}(r_k \mid r_{<k}, c) \;\propto\; p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c) \left(\frac{p_\theta^{\mathrm{CFG}}(r_k \mid r_{<k}, c)}{p_\theta^{\mathrm{CFG}}(r_k \mid \tilde{r}_{<k}, c)} \right)^{\lambda}. \tag{25}$$

In logits, with $g_k^{\mathrm{gen}} = g_k(r_{<k})$ and $g_k^{\mathrm{corr}} = g_k(\tilde{r}_{<k})$, this is $g_k^{\mathrm{gen}} + \lambda(g_k^{\mathrm{gen}} - g_k^{\mathrm{corr}})$, the rule of Eq. (13). The sequential form matches the implementation. The conditional and null evaluations of the corrupted branch share the same corrupted prefix.

APG. The official InfinityStar sampler replaces CFG with adaptive projected guidance (APG) (Sadat et al., 2025). We apply APG to each branch exactly as the official sampler applies it to the generated prefix, which gives g_k^{gen} and g_k^{corr}, and then use Eq. (13) unchanged.

B Implementation details

B.1 Models, samplers and hyperparameters

Tab. 7 lists every setting used for the main results. VAR is evaluated with the ADM suite (Dhariwal & Nichol, 2021), which reports FID (Heusel et al., 2017), Inception Score (IS) (Salimans et al., 2016), and precision and recall (Kynkäänniemi et al., 2019). VPG is added to each model's released sampler, and no base setting is changed (Sec. 5.1).

Corruption. The construction follows Sec. 4.2. Each prefix segment is corrupted once, when it enters the corrupted branch, and reused through the key–value cache. Donors are drawn uniformly with replacement from the same scale. In Infinity and InfinityStar, which encode positions with rotary embeddings rather than an additive embedding, a selected site takes both the donor's input feature and the donor's rotary position, so the replaced input is again a complete code–position pair. The corruption uses its own random generator; CFG and VPG runs with the same seed therefore share every sampling random number, and their difference is a paired effect.

Hyperparameters. *VAR.* The corruption fraction is fixed to n_p=0.1 at every depth; only λ changes with model size (Fig. 5), and d36 uses the d30 value. On seeds 43–46 alone, VPG reduces FID by 0.39 on average ($\Delta_{43\text{–}46}$ in Tab. 2; App. C.1). *Infinity.* We use n_p=0.05 and λ=0.8; on seeds 43–46 alone the DPG-Bench gain is $+0.53$. *InfinityStar.* We use the corruption fraction n_p=0.05 of Infinity and λ=0.5.

Baseline strength. Our CFG baselines match or improve on the FIDs in the official VAR repository (3.37, 2.71, 2.16, 1.97 and 2.65 against 3.55, 2.95, 2.33, 1.97 and 2.63 for d16–d36). The official CFG scale is also close to the FID optimum: re-sweeping it on VAR-d16 (Tab. 8) gives at best 3.34, whereas VPG at the official scale reaches 2.72, so the gain cannot be obtained by retuning CFG.

Table 7: **Settings of the main results.** Base-sampler settings are the official ones released by the authors of each model. The coarse-scale schedules are analyzed in App. C.2.

	VAR (Tab. 2)	Infinity-2B (Tab. 3)	InfinityStar-8B (Tab. 4)
Base sampler (official)			
Task	ImageNet class-conditional	text-to-image	text-to-video
Resolution	256^2 (d16–d30), 512^2 (d36)	1024^2 (1M pixels)	720p
Guidance	CFG 1.5, linear schedule	CFG 4	APG, scale 34
Sampling	top-k 900, top-p 0.96	τ=1	τ_{video}=0.4
VPG			
Corruption fraction n_p	0.1	0.05	0.05
Guidance scale λ	3.2 / 2.3 / 1.8 / 1.3 / 1.3	0.8	0.5
Guided scales	all 10	all	first 12 of 15 (first repeat)
Donor position	learned embedding	rotary (2D RoPE)	rotary (RoPE)
Evaluation			
Benchmark	50k samples, ADM suite	DPG-Bench, 1,064 prompts	VBench, 946 prompts
Seeds	42–46	42–46	0

Table 8: **CFG scale sweep on VAR-d16** (seed 42, 50k samples, official sampler otherwise). VPG uses the official scale 1.5.

CFG scale	1.0	1.25	1.5 (official)	1.75	2.0	2.5	3.0	1.5 + VPG
FID↓	3.61	3.34	3.40	3.67	4.22	5.56	7.04	**2.72**
Precision↑	0.816	0.834	0.849	0.861	0.873	0.891	0.902	0.813
Recall↑	0.537	0.514	0.500	0.494	0.473	0.439	0.420	0.551

B.2 Assets, licenses and compute

VPG is training-free and uses no new data, model weights or annotations. All experiments use released checkpoints and public evaluation assets: the ADM ImageNet reference batch, DPG-Bench and VBench. VAR, Infinity and InfinityStar are released under the MIT License and VBench under Apache-2.0; we use them for research evaluation only and do not redistribute third-party data or weights. Sampling and evaluation ran on NVIDIA H100 and H200 GPUs; the hardware of each cost measurement is stated in App. C.2, because wall-clock comparisons depend on it.

C Additional quantitative results

C.1 Multi-seed results

VAR. Every CFG and VPG entry of Tab. 2 is the mean of five 50k-sample runs with seeds 42–46 (Tab. 9). VPG lowers FID in all 25 paired runs, and no paired 95% interval includes zero. The gain on seeds 43–46 alone (0.39 on average) is almost the same as on all five (0.40). VPG also raises recall at every depth, while precision falls slightly on d16–d30 and rises on d36. VPG thus improves coverage of the data distribution (App. D.4), the opposite of the attention-perturbation references in App. D.2, which raise precision and lower recall.

Infinity. Tab. 10 lists the per-seed DPG-Bench scores behind Tab. 3 (official ELLA scoring over 1,064 prompts). VPG improves every seed, and the 95% interval of the gain excludes zero. Our CFG baseline (83.03) is below the 83.46 reported by Han et al. (2025) because the protocols differ: we score one native 1024^2 image per prompt with a fixed per-prompt seed, and the same sampler scores 83.16 under a four-image 512^2 grid protocol. We therefore compare VPG only with our own baseline.

Table 9: **Multi-seed VAR results.** Five paired 50k-sample runs per depth (seeds 42–46); VPG lowers FID in all 25 pairs. FID: mean ± sample standard deviation. Δ: mean paired difference VPG−CFG, computed from unrounded values, with its two-sided Student-t 95% interval; Rel.: Δ over the CFG mean. $\Delta_{43\text{–}46}$: the same difference on the four seeds not used to select λ. Precision and recall: five-seed means.

	FID↓						Precision↑		Recall↑	
Model	CFG	VPG	Δ	95% CI	Rel.	$\Delta_{43\text{–}46}$	CFG	VPG	CFG	VPG
d16	3.37 ± 0.03	$\mathbf{2.74 \pm 0.03}$	-0.622	$[-0.67, -0.57]$	-18.5%	-0.61	0.850	0.813	0.501	**0.553**
d20	2.71 ± 0.02	$\mathbf{2.26 \pm 0.02}$	-0.452	$[-0.50, -0.41]$	-16.7%	-0.45	0.837	0.809	0.550	**0.581**
d24	2.16 ± 0.03	$\mathbf{1.84 \pm 0.02}$	-0.315	$[-0.37, -0.26]$	-14.6%	-0.30	0.824	0.806	0.584	**0.607**
d30	1.97 ± 0.02	$\mathbf{1.84 \pm 0.02}$	-0.121	$[-0.15, -0.09]$	-6.2%	-0.12	0.816	0.813	0.595	**0.603**
d36	2.65 ± 0.01	$\mathbf{2.17 \pm 0.02}$	-0.484	$[-0.51, -0.46]$	-18.2%	-0.49	0.817	**0.821**	0.556	**0.573**

Table 10: **Multi-seed DPG-Bench results on Infinity-2B.** Official sampler with and without VPG (n_p=0.05, λ=0.8), one native 1024^2 image per prompt; CFG and VPG share the per-prompt seed. Δ: paired difference VPG−CFG with its two-sided Student-t 95% interval.

Seed	42	43	44	45	46	Mean ± s.d.
Infinity-2B (official sampler)	82.97	83.15	83.00	83.07	82.95	83.03 ± 0.08
+ VPG	**84.03**	**83.88**	**83.34**	**83.58**	**83.46**	$\mathbf{83.66 \pm 0.29}$
Δ	+1.07	+0.74	+0.34	+0.51	+0.51	+0.63, 95% CI [0.29, 0.98]

C.2 Compute cost and the coarse-scale schedule

Cost. With CFG, each prefix branch is evaluated with and without the condition, so all-scale VPG runs four Transformer passes per scale instead of two and doubles the Transformer FLOPs (Tab. 11). Most of this cost lies in the finest scales, which hold most tokens. A *coarse-scale schedule* therefore evaluates the corrupted branch only at the coarse scales and runs CFG alone at the rest: on VAR it disables VPG at the three finest of the ten scales, and on InfinityStar it applies VPG at the first 12 of the 15 spacetime scales and only in the first repeat of each scale, leaving later repeats on the clean key–value cache. FLOPs count the Linear, QKV and attention matrix multiplications at 2 FLOPs per multiply–accumulate, and exclude normalization, elementwise operations, sampling bookkeeping and VAE decoding. The VAR timer covers the sampling loop, VAE decoding and host transfer at the batch size of the 50k-sample evaluation.

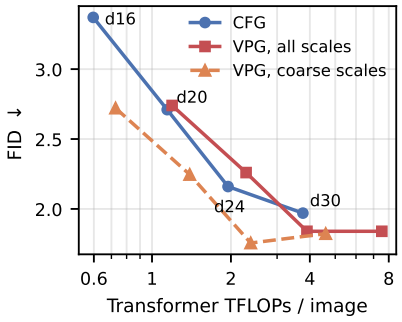


Figure 7: **FID versus compute on VAR** (256^2, Transformer FLOPs per image). Compare CFG, all-scale VPG and coarse-scale schedule.

Table 11: **Cost of VPG.** VAR-d16 (n_p=0.1, λ=3.2): one NVIDIA H100 80GB, batch size 50, 1,000 images after warm-up. InfinityStar-8B (n_p=0.05, λ=0.5): one 720p clip on one NVIDIA H200 with a resident key–value cache. Ratios are relative to the base sampler in the same profile.

Method	Latency↓	Transformer FLOPs↓	Peak memory↓
VAR-d16, per 1k images / per image			
CFG	8.30 s	0.601 TFLOPs	10.30 GiB
+ VPG, coarse-scale schedule	8.91 s (1.07×)	0.733 TFLOPs (1.22×)	10.63 GiB
+ VPG, all scales	11.65 s (1.40×)	1.201 TFLOPs (2.00×)	17.15 GiB
InfinityStar-8B, per clip			
Official sampler (APG)	55.65 s	20.31 PFLOPs	59.44 GiB
+ VPG, coarse-scale schedule	59.37 s (1.07×)	21.76 PFLOPs (1.07×)	77.29 GiB
+ VPG, all scales	107.11 s (1.92×)	40.61 PFLOPs (2.00×)	100.81 GiB

Model	CFG	VPG, all (λ)	VPG, coarse (λ)
VAR-d16	3.40	**2.72** (3.2)	**2.72** (4.0)
VAR-d20	2.71	**2.25** (2.3)	**2.25** (3.6)
VAR-d24	2.20	1.82 (1.8)	**1.76** (3.0)
VAR-d30	1.96	1.85 (1.3)	**1.82** (1.3)
FLOPs	1.00×	2.00×	1.22×

Table 12: **Coarse-scale schedule on VAR** (seed 42, 50k samples). FID of CFG, all-scale VPG and the coarse-scale schedule, with the guidance scale λ in parentheses and the Transformer FLOPs relative to CFG.

Quality of the schedule on VAR. Tab. 12 and Fig. 7 compare the schedule with all-scale VPG. At the all-scale λ the schedule already keeps most of the gain (d16, five seeds: 2.88 ± 0.02, against 3.37 for CFG and 2.74 for all-scale VPG). Because it guides fewer scales, it uses a larger λ. With it, the schedule matches all-scale VPG on d16, d20 and d30 and improves on it on d24, at 1.22× instead of 2× the CFG FLOPs. The d24 improvement holds on further seeds: seeds 43 and 44 give 1.77 and 1.78, against a five-seed mean of 1.84 for all-scale VPG. The schedule lies below the CFG curve at every depth; that is, it reaches a lower FID than CFG at the same compute (Fig. 7).

Schedule and corruption fraction on InfinityStar. Video benefits from a gentler perturbation of the prefix than class-conditional images. InfinityStar uses the small corruption fraction of Infinity (n_p=0.05, half the VAR value), and Tab. 13 shows that the extent of the perturbation matters as well: applying the corrupted branch at all scales and repeats leaves VBench essentially unchanged (83.91 against 83.86). The gain appears when the branch is restricted to the coarse scales, which is also the cheapest setting (1.07× latency).

Table 13: **Where to apply VPG on InfinityStar-8B.** Full VBench (946 prompts, seed 0, n_p=0.05, λ=0.5); latency relative to the official sampler on one H200 (Tab. 11).

Method	Guided scales	Overall↑	Semantic↑	Latency↓
InfinityStar (official sampler)	–	83.86	82.74	1.00×
+ VPG	all 15, all repeats	83.91	82.69	1.92×
+ VPG, coarse-scale schedule (ours)	first 12, first repeat	**84.22**	**83.65**	1.07×

D Analysis of the corrupted-prefix reference

Sec. 5.4 shows that, of four ways to build the corrupted prefix, only same-scale full-embedding replacement improves on CFG. This section formalizes the four rules and compares their distance to the prefix marginal (App. D.1), tests weak references that are not corrupted prefixes at all (App. D.2), describes how the resulting guidance direction differs from CFG scale by scale (App. D.3), and relates the FID response to λ to precision and recall (App. D.4). The FID results of App. D.1–D.2 use VAR-d16 with the official CFG 1.5, seed 42 and 50k samples, for which CFG alone gives 3.40.

D.1 Replacement rules

We use the notation of Sec. 4.2: $e_{j,u} = \mathrm{Emb}(\bar{F}_j[u]) + \mathrm{PosEmb}(j, u)$ at scale j and site u. A fraction n_p=0.1 of prefix sites is selected, the same-scale rules draw a donor site u' from the same scale, and unselected sites are unchanged. The rules test three requirements for the reference: it should remove prefix evidence, stay within the model's same-scale input distribution, and keep every replaced input a consistent code–position pair.

Rule	Replaced input $\tilde{e}_{j,u}$	Tests
Random codebook	$\mathrm{Emb}(z) + \mathrm{PosEmb}(j, u)$, random code z	weak, off-distribution input
Same-scale token	$\mathrm{Emb}(\bar{F}_j[u']) + \mathrm{PosEmb}(j, u)$	content replaced, position kept
Same-scale position	$\mathrm{Emb}(\bar{F}_j[u]) + \mathrm{PosEmb}(j, u')$	position replaced, content kept
Full embedding (ours)	$\mathrm{Emb}(\bar{F}_j[u']) + \mathrm{PosEmb}(j, u') = e_{j,u'}$	complete donor pair

Table 14: **FID, precision and recall along the λ sweep on VAR-d16** (n_p=0.1, 50k samples); the CFG column is λ=0.

λ	CFG	0.4	0.8	1.2	1.6	2.0	2.4	2.8	3.2	3.6	4.0
FID↓	3.40	3.54	3.53	3.45	3.29	3.05	2.87	2.77	**2.72**	2.86	3.09
Precision↑	0.849	0.862	0.867	0.866	0.858	0.850	0.838	0.828	0.810	0.797	0.781
Recall↑	0.500	0.482	0.482	0.493	0.492	0.512	0.530	0.542	0.553	0.566	0.570

Fig. 8 plots FID and IS against λ for the four rules of Tab. 5. The three partial or off-distribution rules reach their best FID at the smallest λ and degrade quickly as λ grows. Full-embedding replacement is the only rule whose FID keeps falling with λ, down to 2.72.

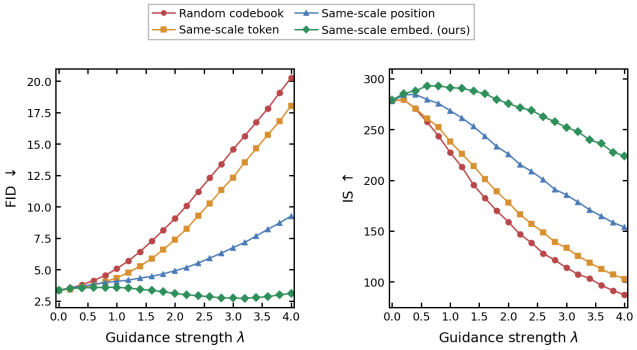


Figure 8: **FID and IS against λ for the four replacement rules** (VAR-d16, n_p=0.1).

Distance to the prefix marginal. We also measure how each rule moves a single prediction head relative to the prefix marginal of App. A.3. For eight VAR-d16 classes, we sample four target rollouts per class without CFG, freeze them, and replay each with one corruption draw per rule (n_p=0.1). Every head is compared with a held-out Monte Carlo estimate B of the marginal, built from 32 prefixes per class that are disjoint from the targets.

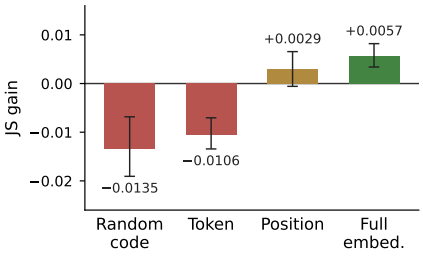


Figure 9: **JS gain against the marginal B.**

Fig. 9 reports the JS gain, the decrease in Jensen–Shannon divergence to B from the clean head to the corrupted head, averaged over sites and scales 2–10; a positive value means that the corruption moves the head toward the marginal. Full-embedding replacement is the only rule whose 95% interval (paired bootstrap over classes) lies entirely above zero, whereas random-codebook and token replacement move the head away. The only rule that moves toward the marginal is thus also the only one that improves FID (Tab. 5). The shift is small (JS to B falls from 0.213 to 0.207), consistent with Sec. 4.1, which treats the corrupted prefix as a local reference rather than an estimate of the marginal.

D.2 Weak references that are not corrupted prefixes

Tab. 15 replaces the corrupted prefix with other weak references: zeroed prefix sites, perturbed attention, a flattened copy of the clean prediction, and a random direction. None improves on CFG.

Table 15: **Weak references that are not a corrupted prefix** (VAR-d16, 50k samples). Best FID of each reference with its λ in parentheses, and precision and recall at that λ.

Weak reference	Best FID↓ ($\lambda^\star$)	Precision	Recall
CFG only (no weak reference)	3.40	0.849	0.500
Drop sites (zero embedding, masked keys)	3.81 (1.8)	0.807	0.523
PAG-style identity attention (2 middle blocks)	4.45 (0.5)	0.892	0.454
SEG-style query blur (σ=2)	7.64 (1.0)	0.867	0.354
Heat 10% of sites, entropy-matched (no extra pass)	4.00 (3.0)	0.867	0.469
VPG direction → norm-matched random	3.40 (1.0)	0.828	0.510
Same-scale full embedding (VPG)	**2.72** (3.2)	0.813	**0.551**

Dropped sites. Zeroing the selected embeddings removes their evidence but replaces it with an input the model never sees in training. It is worse than CFG at every λ (best 3.81), in line with the requirement of Sec. 4.2 that every input remain a valid embedding. *Attention perturbation.* A PAG-style reference that replaces self-attention by the identity in two middle blocks (Ahn et al., 2024), and a SEG-style reference that blurs the attention queries (Hong, 2024), weaken the network rather than the prefix. Both sharpen the samples (precision up to 0.89, recall down to 0.35–0.45) and worsen FID. *Entropy-matched heating.* The corrupted branch predicts a slightly flatter distribution than the clean one (App. D.3). Heating the clean logits at 10% of the sites until their entropy matches, without a second forward pass, gives 4.00, so VPG does not work by extrapolating away from a flatter copy of its own prediction. *Random direction.* Replacing the VPG direction $g_k^{\text{gen}} - g_k^{\text{corr}}$ of Eq. (13) by a random vector of the same norm leaves FID unchanged at λ=1 (3.40) and degrades it at larger λ (5.05 and 12.93), so the gain comes from the direction, not from its magnitude.

Only the corrupted prefix improves on CFG, and it also raises recall the most (0.500 to 0.551). Together with Tab. 5, these results indicate that a useful reference keeps the prefix's own same-scale embeddings as valid inputs while removing part of their evidence.

D.3 Per-scale statistics of the guidance direction

We record per-site statistics of the two branches and of the final VPG logits at every scale of VAR-d16 (100 classes $\times$ 20 samples, n_p=0.1; Fig. 10). *(i) VPG is not a rescaled CFG.* The cosine between the VPG direction $g^{\text{gen}} - g^{\text{corr}}$ and the CFG direction $\ell^{\text{c}} - \ell^{\emptyset}$ is only 0.17–0.23 at every scale, so the two contrasts carry largely different information, as the separate axes of Tab. 1 suggest. *(ii) The reference is nearly as sharp as the clean prediction.* Corrupting the prefix raises the per-site entropy by only 0.05–0.27 nats. VPG therefore exploits *which* tokens lose support when part of the prefix evidence is removed, consistent with the failure of entropy-matched heating (App. D.2). *(iii) Each site is sharpened toward its own prefix.* At λ=3.2 the per-site entropy falls from $\approx$6.5 to 2.5–5 nats, and the fraction of sites whose most likely token changes grows toward the finer scales. Because the favored tokens depend on each sample's own coarse structure, per-site sharpening coexists with the higher recall across samples (Tab. 9).

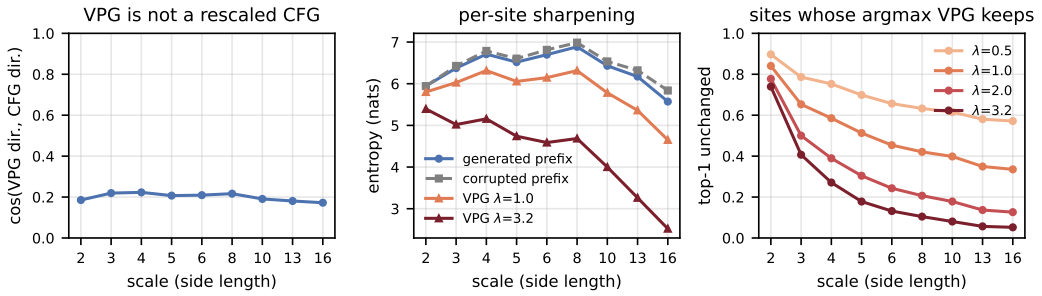


Figure 10: **Per-scale statistics of VPG on VAR-d16.** Left: cosine between the VPG and CFG directions (λ=3.2). Middle: mean per-site entropy under the generated prefix, the corrupted prefix and the final VPG logits. Right: fraction of sites whose most likely token VPG leaves unchanged.

D.4 Precision, recall and diversity

Precision and recall along the VAR-d16 sweep (Tab. 14) clarify the FID curve in Fig. 5. At small λ, VPG mildly sharpens samples, slightly raising precision at the expense of recall, and FID stays near the baseline. At moderate λ, recall increases steadily and the FID improvement tracks this gain in coverage, reaching its best value at the operating point of Tab. 2; as with other guidance scales, much larger λ trades away too much precision. The gain of VPG thus comes mainly from better coverage, consistent with its higher recall at every depth (Tab. 9).

More diversity alone does not explain the gain. A lower CFG scale (Tab. 8) and a higher sampling temperature (FID 3.18, 3.74 and 4.78 at T=1.1, 1.2 and 1.3) also raise recall. Fig. 11 starts all three

knobs from the official sampler: at matched recall VPG reaches a lower FID than either, for example 2.77 against 3.18 at recall 0.54 (λ=2.8 against T=1.1).

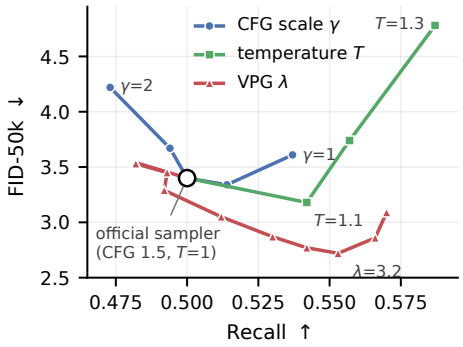


Figure 11: **Three ways to raise recall** (VAR-d16, seed 42, 50k samples), each started from the official sampler: the CFG scale γ (Tab. 8; $\gamma \geq 2.5$ lies above the plotted range), the sampling temperature T, and VPG's λ (Tab. 14).

Table 16: **Prefix branching on VAR-d16** (λ=3.2; 50 classes $\times$ 16 prefixes $\times$ 4 continuations). Mean cosine distance of DINOv2 ViT-L/14 features between continuations of the same prefix (Within) and between samples with different prefixes (Across); Share = Within / Across. Each cell: CFG / VPG.

Prefix up to	Within	Across	Share
side 4	0.277 / 0.282	0.487 / 0.530	0.57 / 0.53
side 6	0.214 / 0.215	0.488 / 0.533	0.44 / 0.40
side 8	0.149 / 0.144	0.488 / 0.535	0.30 / 0.27

Where the diversity enters. We fix a prefix and resample only the later scales (Tab. 16). For 50 classes, 16 prefixes are generated up to side 4, 6 or 8, and each is continued four times with fresh sampling draws; the corruption draws are held fixed, so that their randomness is not counted as diversity. VPG raises the distance between samples with different prefixes by 9–10% but leaves the distance between continuations of the same prefix unchanged, so the extra diversity is already present in the prefix up to side 4, the first four of ten scales. This agrees with App. D.3 (iii): each site is sharpened toward its own prefix, so a given prefix is continued no more diversely, while different samples differ more.

E Limitations and Broader Impacts

Limitations. VPG is an inference-time rule and complements, rather than replaces, training-time remedies that change the learned dynamics of the generator. Its two hyperparameters play the role of the CFG scale: we fix n_p per model family and set λ per model, and the best λ decreases with model size. The gain is also smaller for the largest 256^2 VAR model (0.12 FID on d30) than for the others. All-scale VPG doubles Transformer FLOPs and raises peak memory, so the coarse-scale schedule, at 1.22$\times$ FLOPs on VAR and 1.07$\times$ latency on InfinityStar, is the practical setting; on InfinityStar.

Broader impacts. VPG improves released image and video generators without retraining, which can save training compute when a released checkpoint is otherwise adequate. The same improvement can make synthetic media more convincing for deception, impersonation or disinformation. Because VPG only changes the sampler of an existing model, it inherits that model's safeguards, such as access control, provenance marking, usage monitoring and safety filtering, and should be deployed under the same standards as the generator it modifies.

F Qualitative Results

This section complements Sec. 5.3. It explains the teaser (App. F.1) and adds paired comparisons for VAR, Infinity and InfinityStar (App. F.2–F.4). Every pair uses the same checkpoint, condition, seed and base sampler; only the VPG branch is switched on or off.

F.1 TEASER FIGURE

Left. The first two columns use Infinity (Han et al., 2025). The first prompt asks for a wax seal embossed with the letters "VPG": without VPG the seal reads closer to "VDG", and VPG renders the intended letters. The second prompt describes an owl standing among shattered mirror pieces on a floor; the baseline loses the owl in the broken glass, and VPG keeps the subject. The third column uses VAR-$d30$ (Tian et al., 2024) on the ImageNet (Deng et al., 2009) class "macaw": the baseline produces an occluded, malformed bird, and VPG a complete scarlet macaw. The last two rows use InfinityStar (Liu et al., 2025) with the prompt "A towering moss-covered mech kneeling to tend a tiny rooftop bonsai garden during light rain, steam mixing with mist, tactile metal textures, contemplative mood, ultra-realistic." VPG keeps the mech, the bonsai and the rooftop consistent across frames.

Right. The two guidance rules act on different inputs of $p_\theta(r_k \mid r_{<k}, c)$. CFG contrasts the condition c with the null condition $\emptyset$ at a fixed prefix and favors evidence for the condition, $p(c \mid r_{\leq k})^\gamma$. VPG contrasts the generated prefix $r_{<k}$ with a corrupted copy $\tilde{r}_{<k}$ at a fixed condition and favors candidates supported by the generated prefix (Eq. (8)). A frozen visual AR model has no null-prefix mode, so the prefix cannot simply be dropped as the condition is in CFG; the corrupted prefix provides the reference instead (Sec. 4.1).

F.2 VAR COMPARISONS

Fig. 12 compares CFG and CFG + VPG on VAR-$d36$ (512^2) at the operating point of Tab. 2 (n_p=0.1, λ=1.3) for twelve ImageNet classes. In each panel, the two rows share all sampling random numbers, so each column compares the two samplers from the same start. Across the twelve classes, the differences concern the global layout of the object rather than its texture: where CFG merges two instances, truncates an object or leaves an ambiguous shape, VPG produces a single, complete object, while the background and colors often stay close to the CFG sample. This matches App. D.4, where the extra diversity of VPG enters in the first four of ten scales, which fix the layout.

F.3 DPG-BENCH EXAMPLES

Figs. 13–14 compare the official Infinity-2B sampler with and without VPG (n_p=0.05, λ=0.8) on DPG-Bench prompts, each over four seeds shared by the two methods. As in Fig. 4, the highlighted prompt elements are those that VPG renders more reliably: objects placed in or on others, counts and attributes, and short text. These improvements are consistent across seeds rather than isolated. VPG places the white candle in the bottle's spout in all four seeds, against one of four for the official sampler, and renders the uniformly white squares among the colored triangles more consistently. The main subject of each prompt is already rendered well by the official sampler; the gain concerns the secondary elements, as in Fig. 4.

F.4 INFINITYSTAR COMPARISONS

Fig. 15 compares the official InfinityStar sampler with VPG (n_p=0.05, λ=0.5, coarse-scale schedule) on three prompts, showing five uniformly spaced frames per clip. VPG keeps the prompted concept and scene layout more consistent across frames. The differences again follow the secondary prompt elements. In *subway greenhouse*, VPG covers the train cars themselves with vines and glowing flowers, as the prompt asks, whereas the official sampler places the vegetation mainly along the wall and the platform edge. In *origami whale*, VPG renders the gulls and mist moving through the scene and keeps the tugboats small relative to the whale. These elements persist across all five frames, so the difference is not a single-frame effect. The prompts are: *train library*: "A vintage train car transformed into a traveling library speeding through snowy mountains at sunrise, warm interior lamps, fluttering book pages, parallax through the windows, cinematic realism"; *subway greenhouse*: "An abandoned underground subway platform transformed into a lush greenhouse, vines wrapping around old train cars, bioluminescent flowers pulsing as a train breeze moves the leaves, cinematic realism, rich tactile textures"; *origami whale*: "A colossal origami whale gliding through a foggy harbor at dawn, folded paper surfaces catching golden light, tiny tugboats circling below, gulls and mist moving through the scene, poetic cinematic realism, ultra-detailed."



Figure 12: **Same-seed comparisons on VAR-$d36$ (ImageNet 512^2).** In each panel, top row: CFG (official sampler); bottom row: CFG + VPG (n_p=0.1, λ=1.3) with the same random numbers. VPG turns instances that CFG merges into a single animal (zebra, monarch butterfly, Siberian husky), completes objects that CFG leaves truncated or ambiguous (volcano, hot air balloon, pizza), and repairs distorted anatomy (golden retriever, tabby cat).



Figure 13: **DPG-Bench examples on Infinity-2B.** Four seeds per prompt, shared by the two methods; **highlighted** prompt elements are rendered more reliably with VPG.

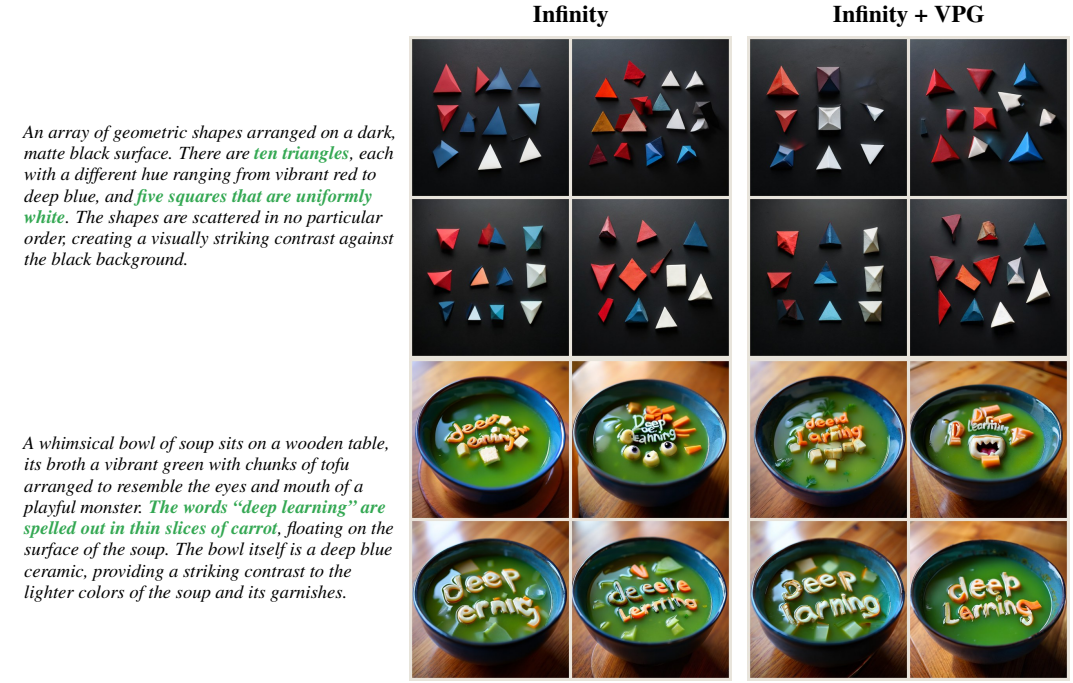


Figure 14: **DPG-Bench examples on Infinity-2B, continued.** Four seeds per prompt, shared by the two methods.

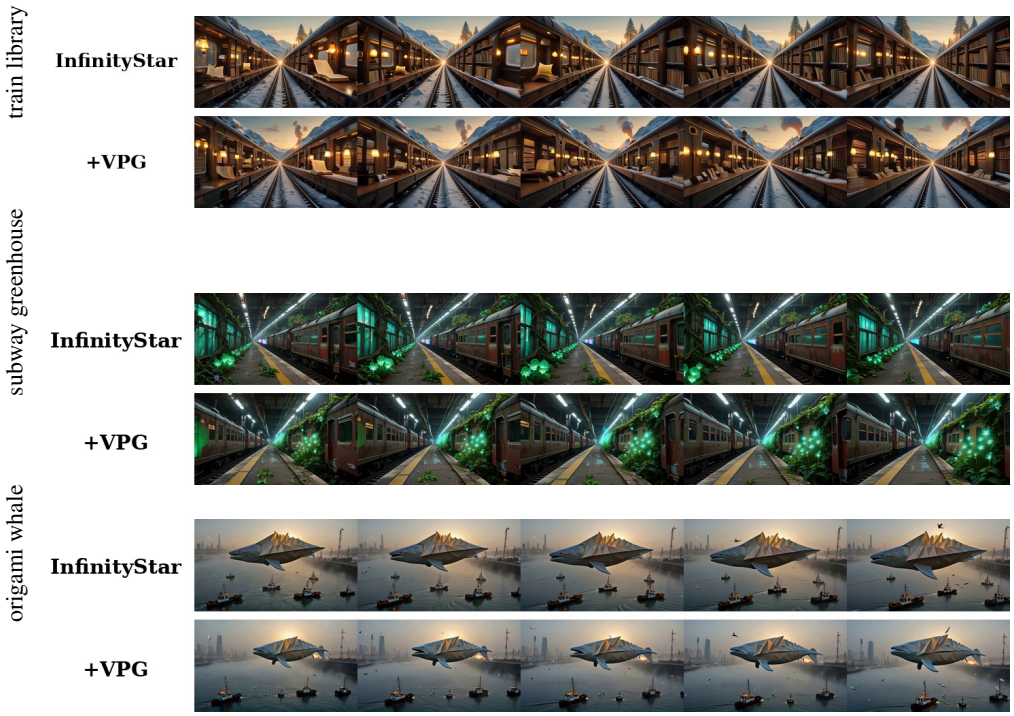


Figure 15: **InfinityStar comparisons.** Each panel shows five frames of the official sampler (top) and of the same seed with VPG (bottom). VPG better preserves the vegetation and bioluminescent cues of the greenhouse, the folded-paper whale and the scale of the harbor.