FREE DIVERSITY FOR EFFICIENT TEST-TIME SCALING IN VISUAL AUTOREGRESSIVE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual Autoregressive (VAR) models generate images by next-scale prediction, and their earliest scales strongly shape the final image. This sensitivity has been taken to imply that these scales must remain text-conditioned. We revisit this assumption and find that text conditioning has little effect at the coarsest scales: sampling them unconditionally leaves mean quality nearly unchanged while substantially diversifying the generated images. Existing diversity-boosting methods for VAR improve best-of-N performance at the cost of either latency or mean quality; in contrast, unconditioning the earliest scales incurs no extra latency and a smaller drop in mean quality while yielding higher diversity and best-of-N performance. Building on this finding, we propose early attention pruning (EAP), an efficient verification method that prunes failing candidates at early scales using cross-attention signals, without decoding intermediate images. Our training-free pipeline outperforms existing diversity-boosting and test-time scaling methods on T2I-CompBench++ and GenEval2 with much less computation, and generalizes across VAR architectures and model sizes.

1 INTRODUCTION

Visual Autoregressive (VAR) models (Tian et al., 2024) generate images through *next-scale prediction*: starting from a coarse 1×1 token map, the model produces progressively higher-resolution maps, generating all tokens within a scale in parallel. This coarse-to-fine hierarchy, adopted by recent text-to-image systems such as Infinity (Han et al., 2025) and Switti (Voronov et al., 2024), yields competitive image quality with substantially faster inference than diffusion models (Ho et al., 2020; Esser et al., 2024; Podell et al., 2024). Because every scale conditions all finer ones, the earliest scales exert a strong influence on the final image. Prior work has accordingly treated them as critical: they have been described as the stage where semantics are established (Li et al., 2025), perturbing them has been shown to significantly degrade reconstruction (Liu et al., 2026), and searching over them has been reported to cause uncontrollable quality degradation (Tang et al., 2025). This sensitivity has often been taken to imply that early-scale text conditioning must be preserved, or that removing it requires additional compensation such as time-traveling (Park et al., 2025).

In this work, we revisit this assumption: sensitivity implies that changes at early scales propagate to the final image, not that such changes are harmful. Probing the hidden states of two text-to-image VAR models with distinct scale hierarchies, Infinity (Han et al., 2025) and Switti (Voronov et al., 2024), we find that object-level attribute binding is weak up to the 2×2 scale. Consistently, removing text conditioning at these coarsest scales changes mean quality only marginally while substantially increasing layout diversity and the oracle best-of-N score, whereas extending it to 4×4, where attributes become localized, removes most of these gains in both models.

These findings make early-scale unconditioning well suited to **test-time scaling**. Oracle best-of-N naturally rises with diversity, but practically a verifier is not an oracle. Thus, when diversity comes at the cost of mean quality, the verifier may select poor candidates (Troshin et al., 2025; He et al., 2025; Kim et al., 2025). Early-scale unconditioning, which we call unconditioned anchors (UA), avoids this trade-off: its diversity is free, requiring no extra compute and costing little mean quality while raising the oracle. The remaining bottleneck is verification: image reward models are unreliable on early-scale decodes (Chen et al., 2025), and VAE decoding adds latency. We sidestep both limitations with cross-attention, building on evidence that the separation of cross-attention maps across prompt

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.

objects reflects object separation and attribute binding (Chefer et al., 2023; Rassin et al., 2023; Guo et al., 2024). We prune compositionally failing candidates at early scales based on cross attention map dissociation, which we call early attention pruning (EAP), and score only the survivors with a reward model at a later scale, yielding verification that is both cheap and accurate.

Our contributions are as follows:

- We show that removing text conditioning at the coarsest VAR scales leaves mean quality nearly unchanged while substantially diversifying the outputs, raising the oracle in every subtask and most on structure-dependent ones such as spatial relations and shape. Using linear probing and causal interventions, we identify the scale at which conditioning becomes necessary.
- We propose a training-free test-time scaling pipeline that pairs UA, which diversifies candidates at no extra compute, with EAP, which prunes compositional failures from cross-attention before external verifiers become reliable, cutting test-time compute.
- We validate our method on T2I-CompBench++, GenEval2, and Pick-a-Pic, where it outperforms existing diversity-boosting and test-time scaling methods with significantly less computation, and generalizes across Infinity-2B, Infinity-8B, and Switti.

2 Related Work

Visual autoregressive generation. Early AR image models generate pixels or tokens in raster-scan order (Van Den Oord et al., 2016; Salimans et al., 2017; Van den Oord et al., 2016); vector quantization (Van Den Oord et al., 2017; Esser et al., 2021; Razavi et al., 2019) enabled scaling to text-to-image generation (Ramesh et al., 2021; Wang et al., 2024), but token-by-token decoding remains slow. VAR (Tian et al., 2024) instead predicts all tokens of a scale in parallel, and subsequent work adds CLIP-based text conditioning (Zhang et al., 2024), hybrid tokenization (Tang et al., 2024), non-causal transformers (Voronov et al., 2024), and bitwise tokenization (Han et al., 2025).

Guidance and diversity. Classifier-free guidance (Ho & Salimans, 2022) improves prompt adherence at the cost of diversity. Diffusion models mitigate this by restricting guidance to a limited noise interval (Kynkäänniemi et al., 2024) or annealing noise in the condition (Sadat et al., 2024), but the analogous question is underexplored in VAR, whose discrete token maps commit the global layout within a few forward passes.

Compositional generation and cross-attention. Both diffusion (Rombach et al., 2022; Peebles & Xie, 2023; Podell et al., 2024) and VAR models often fail on compositional prompts (Huang et al., 2023; 2025; Ghosh et al., 2023; Kamath et al., 2025), motivating preference optimization (Lee et al., 2025), composable conditioning (Liu et al., 2022), and inference-time interventions (Park et al., 2024; Yu & Gao, 2025). Several of these treat cross-attention maps as a proxy for compositional correctness, using them as an optimization objective (Chefer et al., 2023; Rassin et al., 2023) or a noise-selection criterion (Guo et al., 2024). We instead use them as a near-zero-cost *verifier* at early VAR scales, pruning candidates before most tokens are generated.

Test-time scaling. Test-time scaling samples multiple candidates and selects among them with verifiers (Ma et al., 2023; Guo et al., 2025; Snell et al., 2024); its gains depend on diversity (Chung et al., 2025), which typically lowers mean quality (Troshin et al., 2025; He et al., 2025; Kim et al., 2025). Verifier-guided search scales well for diffusion (Ma et al., 2025; Xie et al., 2025) but remains underexplored for VAR, where existing approaches rely on many samples (Chen et al., 2025; Tang et al., 2025) or quality-reducing noise injection (Park et al., 2025) for diversity, and on decoded previews that are costly and poorly scored for intermediate verification (Chen et al., 2025); UA and EAP address these two limitations, respectively.

3 Preliminaries and Observations

3.1 Preliminaries

Next-scale prediction. Conventional autoregressive (AR) image generators (Van Den Oord et al., 2016) predict an image token by token in raster-scan order. Visual Autoregressive modeling

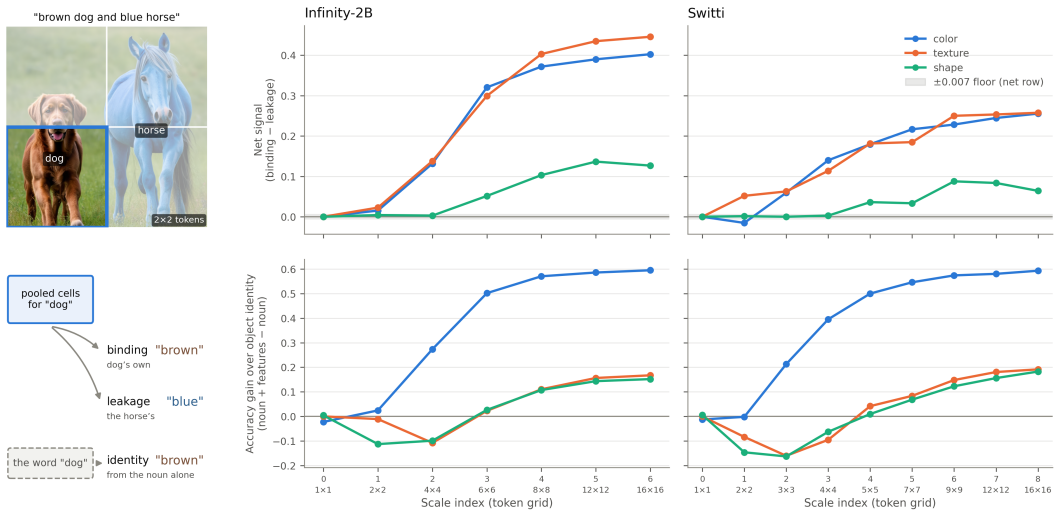


Figure 1: **Attribute localization probing. Left:** pooled features of the tokens under an object's mask (here at 2×2) predict its own attribute (*binding*) and the other object's (*leakage*); a third probe sees only the noun. **Right:** net signal, binding − leakage (top), and the accuracy gain of a probe given both the noun and the features over the noun-only probe, i.e., attribute information beyond object identity (bottom), for Infinity-2B and Switti. Full values in section A.

(VAR) (Tian et al., 2024) instead takes *multi-scale token maps* as the autoregressive unit. A visual encoder maps an image to a feature map $f \in \mathbb{R}^{h\times w\times C}$, which is residually quantized into L token maps of increasing resolution, $\mathcal{R} = (r_0, r_1, \ldots, r_{L-1})$, where r_s has spatial size $h_s \times w_s$ and encodes only the information not yet explained by coarser scales. We index scales from $s{=}0$ (the $1{\times}1$ map) to $s{=}L{-}1$. VAR factorizes the joint distribution scale by scale as $p(r_0, \ldots, r_{L-1}) = \prod_{s=0}^{L-1} p(r_s \mid r_{<s}, c)$, where c is the text condition, predicting all $h_s \times w_s$ tokens of r_s in parallel. Generating an image thus takes only L forward passes.

Classifier-free guidance. Text-to-image VAR models apply classifier-free guidance (CFG) (Ho & Salimans, 2022) at every scale. Let ℓ_s^c and $\ell_s^\varnothing$ denote the logits for r_s under the text condition and the null condition, respectively. Tokens are sampled from

$$\ell_s \;=\; \ell_s^\varnothing + w_s\big(\ell_s^c - \ell_s^\varnothing\big), \tag{1}$$

where w_s is the guidance weight at scale s, and $w_s{=}0$ corresponds to purely unconditional generation.[1]

3.2 Do Early Scales Need Text Conditioning?

Despite their sensitivity (section 1), whether early scales need text conditioning should depend on what information each scale encodes; we test this correlationally by probing where attribute information is localized (section 3.2.1) and causally by removing text conditioning at early scales (section 3.2.2).

3.2.1 Where Are Attributes Localized?

Setup. We extract scale-wise representations by teacher-forcing the visual tokens of synthetic images into Infinity-2B and Switti and taking the hidden states at each scale. We construct 3.5K image–prompt pairs from T2I-CompBench++ (Huang et al., 2023), filtered following the VSC pipeline (Dat et al., 2025), and extract object masks with SAM3 (Carion et al., 2025). The full protocol is in section A.1.

For each object, we train a linear probe on the tokens covered by its mask to predict the object's own attribute (*binding*) and the attribute of another object in the same image (*leakage*); in fig. 1 (left), the

[1] Switti parameterizes CFG as $\ell_s^c + w'_s(\ell_s^c - \ell_s^\varnothing)$; $w_s{=}0$ in eq. (1) corresponds to $w'_s{=}{-}1$.

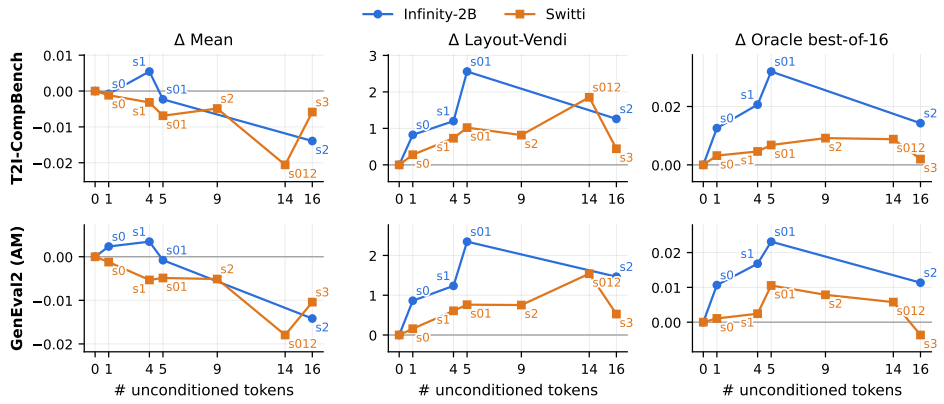


Figure 2: **Removing text conditioning at the coarsest scales barely changes mean quality while increasing diversity and oracle performance.** Changes in mean score (left), Layout-Vendi (middle), and oracle best-of-16 (right) relative to the fully conditioned baseline, on T2I-CompBench++ (top) and GenEval2 (bottom). The x-axis is the number of unconditioned tokens; point labels indicate the unconditioned scales (e.g., s01 is Uncond@$\{0, 1\}$). Full numbers are in section B.

pooled 2×2 cells of the dog in "brown dog and blue horse" should predict "brown" but not "blue". We report the difference in probe accuracy as the *net signal* (top row of fig. 1). A positive net signal indicates that an object's tokens encode which attribute belongs to that object, rather than only which attributes appear in the image.

However, attributes in these prompts are not independent of object identity (a wallet is almost always leather, a die almost always cubic), so a probe that merely recognizes the object predicts its attribute for free. A noun-only probe, which receives the object's noun as a one-hot vector with no visual features, already reaches 0.31 for color, 0.73 for texture and 0.45 for shape. We therefore report the gain of a probe given both the noun and the pooled features over this noun-only probe, which isolates the attribute information beyond object identity (lower row of fig. 1).

Results. The net signal (fig. 1, top) remains near zero at 2×2 for every attribute in both models, even though the objects occupy distinct 2×2 regions of the final image, and rises sharply between 4×4 and 6×6 in Infinity and gradually from 3×3 in Switti.

Controlling for object identity (fig. 1, bottom; values below zero mean the features add nothing beyond the noun) further separates color from the other two attributes: its gain over the noun-only probe is clear from 4×4 in Infinity and 3×3 in Switti, whereas texture and shape do not exceed the noun-only probe until 6×6 in Infinity and 5×5 in Switti (table 10).

These results suggest that the coarsest scales carry little object-level attribute information for the text condition to control. They do still encode information not tied to specific objects, such as global color statistics and coarse layout; we next test causally whether this information needs the text condition.

3.2.2 Removing Text Conditioning at Early Scales

Setup. For a set of scale indices $\mathcal{S}$, **Uncond@**$\mathcal{S}$ sets $w_s{=}0$ in eq. (1) for all $s \in \mathcal{S}$ and keeps the original text-conditioned sampler at all other scales. We evaluate single scales and contiguous windows up to 4×4; for example, Uncond@$\{0, 1\}$ removes conditioning at the 1×1 and 2×2 scales. Because the two models follow different scale schedules, the same index can correspond to different resolutions (4×4 is $s{=}2$ in Infinity and $s{=}3$ in Switti), so fig. 2 places each condition by its number of unconditioned tokens. We evaluate on T2I-CompBench++ and GenEval2 Arithmetic Mean (AM), generating $N{=}16$ images per prompt. For each prompt, we measure three quantities: the *mean* score over the 16 samples, which reflects the quality of a single draw; *diversity*, measured by Layout-Vendi; and the *oracle* best-of-16 score, which bounds what a verifier could select. We report each as a change relative to the fully conditioned baseline, averaged over prompts.

Results. Removing conditioning at 1×1 and 2×2, individually or together, changes the mean only marginally on both benchmarks while increasing both Layout-Vendi and the oracle in both models, more so in Infinity, whose samples are innately less diverse than Switti's (table 11). Extending the

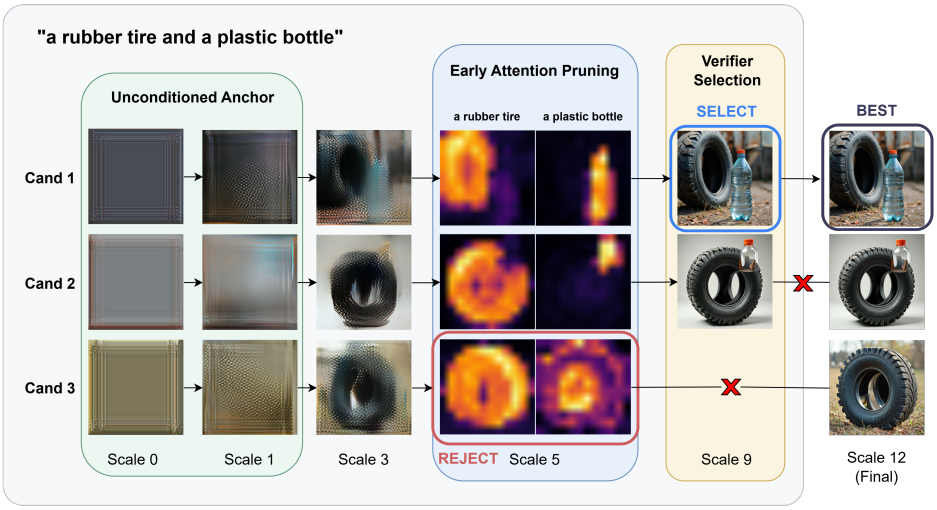


Figure 3: **Overview.** Unconditioned anchors sample the earliest scales of all N candidates without the text condition. The cascade prunes candidates at scale 5 by their cross-attention (early attention pruning, EAP), verifies the survivors on scale-9 previews, and finishes only the winner.

intervention to 4×4, where probing shows that attributes become localized, removes most of this benefit: in Infinity the mean drops faster and the oracle gain shrinks, and in Switti the diversity and oracle gains largely vanish. The models differ because Switti's intermediate 3×3 scale already partially localizes the composition, whereas Infinity has no such scale.

Together with the probing results, these findings indicate that text conditioning becomes necessary once attributes begin to be localized; before that, unconditioning raises diversity and the oracle at no extra compute and little cost to mean quality, a natural basis for test-time scaling.

4 Method

We turn the observation of section 3.2 into a test-time scaling method with two parts (fig. 3): unconditioned anchors widen the candidate pool (section 4.1), and a verification cascade selects from it at low cost (section 4.2).

4.1 Unconditioned anchors

Unconditioned anchors (UA) sample unconditionally on an *anchor set* $\mathcal{A}$ of early scales ($w_s{=}0$ in eq. (1), i.e., Uncond@$\mathcal{A}$) and keep the constant weight w of standard sampling elsewhere. We call these scales *anchors* because every later scale is generated conditioned on them: they fix each candidate's coarse layout, which differs across candidates. Guidance already computes both branches, so UA only changes a weight: it requires no extra compute, no auxiliary model, and no training.

Design choices. We sample the anchors from the model's own unconditional distribution rather than noising the text condition as DiverseVAR does, and with $w_s{=}0$ rather than $w_s{=}1$ (tables 1 and 3). We place them below the scale at which the probing of section 3.2 finds attributes bound to their objects in either model: $\mathcal{A}{=}\{0, 1\}$ (1×1 and 2×2), fixed from the probe rather than tuned per model or benchmark, for both models and all best-of-N experiments.

4.2 Verification cascade

A wider pool raises the *oracle*, the score of its best candidate, but best-of-N selects the verifier's top pick, so the *realized* score of the selected candidate depends on how well the pool is ranked. We make selection stronger without making it more expensive.

For each prompt, the cascade (i) generates all N candidates to scale 5; (ii) prunes the lowest-scoring 30% with *early attention pruning* (EAP), which reads cross-attention the forward pass has already computed; (iii) continues the survivors to scale 9 and scores their decoded previews with a decomposed verifier; and (iv) completes only the winner to the final scale. Unlike a classical cascade with a weaker cheap model, ours applies the same model to an earlier, much smaller version of each

sample: scales 0–5 contain only 2.5% of the tokens and scales 0–9 24%. These scale indices are for Infinity (L=13); for other generators we prune at the scale with the closest token fraction (scale 6 and 2.0% for Switti; section 5.4).

EAP signals. At scale 5, the cross-attention already shows where each word of the prompt is being drawn. We use it to catch two compositional failures: two objects drawn in the same place, and an attribute landing on the wrong object. Each prompt is parsed once with a dependency parser (Qi et al., 2020): the head noun of each noun phrase becomes an object and the words modifying it become its attribute, so "a rubber tire and a plastic bottle" (fig. 3) yields the objects *tire* and *bottle* and the pairs *rubber→tire* and *plastic→bottle*. For a word with text tokens g, its heat map p_g is the scale-5 cross-attention $A[q,t]$ from image position q to token t, averaged over heads, over the early and middle blocks (0–21 of 32; section C.3), and over the tokens in g, then normalized over positions:

$$p_g[q] \;=\; \frac{\bar{a}_g[q]}{\sum_{q'} \bar{a}_g[q']}, \qquad \bar{a}_g[q] \;=\; \frac{1}{|g|}\sum_{t\in g} A[q,t]. \tag{2}$$

The two signals are

$$C \;=\; \min_{i\neq j} \mathrm{JSD}\big(p_{o_i},\, p_{o_j}\big), \qquad S \;=\; \min_{(\alpha\to o)} \sum_{q\in R_o} p_\alpha[q]. \tag{3}$$

C is the Jensen–Shannon divergence between two objects' heat maps, so a low C means two objects share one region. S is the share of an attribute's heat map that falls inside its object's region R_o, taken as the 30% of positions where p_o is largest: about 0.3 if the attribute is spread uniformly, 1 if it lies entirely on the object. Taking the minimum over pairs, rather than the mean, lets one colliding pair or one misplaced attribute mark a candidate even if the others are fine. Prompts with no object pair or attribute ("two boys"; 73 of $2{,}400$ on T2I-CompBench, none on GenEval2) skip pruning.

EAP rule. Since C and S have different ranges, we combine them by rank. Within a prompt, each defined signal is ranked over the N candidates and normalized to $(0, 1]$ (higher is better); a candidate's score u_i is the mean of its normalized ranks, and the lowest $\mathrm{round}(0.3N)$ are pruned:

$$u_i \;=\; \operatorname*{mean}_{X\in\{C,S\}\text{ defined}} \mathrm{rank}_i\big(X\big). \tag{4}$$

EAP only has to find the worst candidates, not to rank the pool, and pruning a fixed count keeps the per-prompt cost fixed.

Verifier. Preference models such as HPSv2 (Wu et al., 2023) and ImageReward (Xu et al., 2023), common in test-time scaling (Ma et al., 2025; Chen et al., 2025), return one aggregate preference score rather than checking what a compositional prompt asks for. We instead adopt decomposed compositional evaluation (Hu et al., 2023; Cho et al., 2024) for selection, with a multimodal LLM that both writes and answers the questions. Once per prompt, the model writes the K yes/no questions the prompt entails, from the text alone. It then answers them on each survivor's scale-9 preview $\tilde{I}_i$, and the candidate's score is the geometric mean $v_i = \big(\prod_{k=1}^{K} P_k(\text{yes} \mid \tilde{I}_i)\big)^{1/K}$, where P_k is read from the first generated token for the k-th question; a preference model can take the verifier's place (section 5.2).

5 Experiments

Setup. We use Infinity-2B (Han et al., 2025) (1024×1024, L=13 scales) on T2I-CompBench++ (Huang et al., 2025) (hereafter T2I-CompBench; 2,400 prompts in eight categories, scored by its official evaluators: BLIP-VQA for attribute binding, UniDet-based detection for spatial relations and numeracy, CLIPScore for non-spatial relations, and their average for complex prompts) and GenEval2 (Kamath et al., 2025) (800 prompts, scored by Soft-TIFA with Qwen3-VL, which answers each prompt's atomic questions; AM and GM are the arithmetic and geometric means of the atom scores, GM approximating prompt-level correctness), with N=16 candidates per prompt unless stated otherwise. Baselines are standard sampling (w=4) with best-of-N, TTS-VAR (Chen et al., 2025), and DiverseVAR (Park et al., 2025), which noises the text condition (*anneal*) and

Table 1: **Pool quality and what one selector recovers from it.** $N{=}16$. Mean: average over the 16 candidates; oracle: best of the 16; realized: the candidate selected by our cascade with the IVL verifier, applied identically to every pool. Best per column in bold, second best underlined.

	T2I-CompBench			GenEval2 GM			GenEval2 AM		
Sampler	mean	oracle	realized	mean	oracle	realized	mean	oracle	realized
Standard ($w{=}4$)	**0.4700**	0.6183	<u>0.5285</u>	<u>0.1614</u>	0.4388	0.3210	**0.6192**	0.7957	0.7136
DiverseVAR (anneal)	0.3816	0.6447	0.5081	0.1004	0.4340	0.3200	0.5150	0.7891	0.6962
DiverseVAR (full)	0.4069	<u>0.6555</u>	0.5255	0.1112	0.4547	<u>0.3366</u>	0.5467	0.8053	0.7165
UA, $\mathcal{A}{=}\{0,1,2\}$ (ablation)	0.4296	**0.6574**	0.5257	0.1305	<u>0.4754</u>	0.3233	0.5888	<u>0.8161</u>	<u>0.7197</u>
UA, $\mathcal{A}{=}\{0,1\}$ (ours)	<u>0.4677</u>	0.6503	**0.5380**	**0.1622**	**0.4826**	**0.3540**	<u>0.6185</u>	**0.8188**	**0.7370**

Table 2: **Main comparison, scored on the selected image.** Rows pair a sampler with a selection rule, $N{=}16$ except the single sample; TTS-VAR is a selection pipeline, run on either sampler. T2I-CB: official score over the eight categories; cost per prompt (wall-clock in s), measured on T2I-CompBench. Best per column within a verifier block in bold, second best underlined.

		Verifier: IR					Verifier: IVL				
		T2I-CB	GenEval2		cost		T2I-CB	GenEval2		cost	
Sampler	Selection	overall	GM	AM	wall	TFLOPs	overall	GM	AM	wall	TFLOPs
Standard	single sample	0.470	0.1614	0.6192	2.3	102	0.470	0.1614	0.6192	2.3	102
Standard	best-of-N	0.517	0.2250	0.6851	<u>22.5</u>	1,643	0.525	0.3297	0.7141	<u>32.5</u>	2,154
DiverseVAR	best-of-N	0.524	0.2170	0.6865	32.4	1,777	0.518	<u>0.3608</u>	0.7223	40.9	2,288
Standard	TTS-VAR	0.503	0.1895	0.6615	34.8	<u>1,369</u>	0.503	0.2720	0.6807	43.0	<u>1,816</u>
UA	TTS-VAR	0.517	0.2005	0.6735	34.7	<u>1,369</u>	0.512	0.2848	0.6893	42.6	<u>1,816</u>
UA	best-of-N	<u>0.529</u>	**0.2338**	**0.6997**	<u>22.5</u>	1,643	<u>0.535</u>	**0.3793**	**0.7439**	33.3	2,154
UA	cascade (ours)	**0.531**	<u>0.2289</u>	<u>0.6944</u>	**7.0**	**452**	**0.538**	0.3540	<u>0.7370</u>	**12.7**	**808**

rewinds generation to an earlier scale (*full*, used unless noted). Best-of-N verifies the finished images; BoN@s9, like our cascade, verifies the scale-9 previews and finishes only the winner. Each method is run with two verifiers, the decomposed verifier of section 4.2 on InternVL3.5-14B (Wang et al., 2025) (IVL) and ImageReward (Xu et al., 2023) (IR); neither is itself a benchmark judge, and both are applied identically to every method. The IVL questions are written by the model from each prompt on T2I-CompBench and decomposed by GenEval2's official logic on GenEval2, and are the same for every method. Cost is per prompt and includes generation, decoding and verification; wall-clock is on one A100-80GB and, unlike TFLOPs, depends on the implementation (the released DiverseVAR code is sequential).

5.1 The candidate pool

Table 1 isolates the sampler. UA raises the oracle on both benchmarks ($+0.032$ on T2I-CompBench) with a negligible change in the mean. Raising the oracle alone is easy: DiverseVAR and the anchor set $\{0,1,2\}$ reach a similar T2I-CompBench oracle, but they lower the mean and, under the same selector, their selected candidates score *below* standard sampling on T2I-CompBench and barely above it on GenEval2. Ours scores the highest on every metric: more bad candidates give the verifier more ways to be wrong.

5.2 Main comparison

UA adds no compute (table 2), whereas DiverseVAR, which re-runs intermediate scales, costs 6–8% more. Under best-of-16, replacing standard sampling by UA raises every score under both verifiers at the same cost and beats DiverseVAR on every column at lower cost; it also improves TTS-VAR, which still scores at or below standard best-of-16. The cascade matches UA best-of-16 at about a third of its compute ($0.28\times$ TFLOPs with IR, $0.38\times$ with IVL): it scores slightly higher on T2I-CompBench and gives back up to 0.025 GenEval2 GM. Across budgets, it matches standard best-of-16 at $4.9\times$ less compute (fig. 4). At equal compute on GenEval2, it scores higher than every best-of-N and TTS-VAR, 0.033 GM above UA best-of-N and 0.066 above standard (fig. 8).

5.3 Ablations

Anchor set and guidance weight. Table 3 varies the anchor set and the guidance weight. No single scale reaches the oracle of $\{0,1\}$; $\{1\}$ raises the mean but scores lower under the cascade.

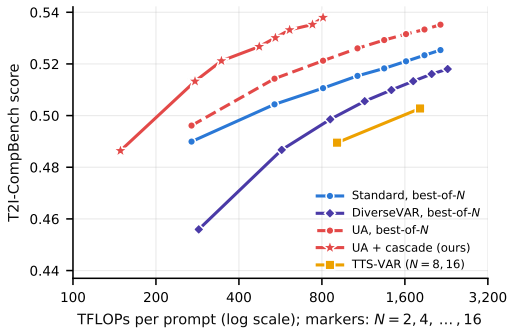


Figure 4: **Quality against compute**, T2I-CompBench, IVL. Markers are $N{=}2, 4, \ldots, 16$ candidates (TTS-VAR: $N{=}8, 16$); the cascade prunes at scale 5 and verifies at scale 9.

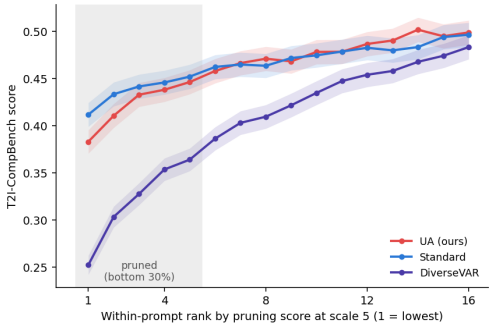


Figure 5: **EAP is a floor cutter.** Mean T2I-CompBench score by within-prompt rank of the EAP score (eq. (4)) at scale 5, with 95% confidence bands; shaded ranks are pruned.

Table 3: **Anchor set and guidance weight.** T2I-CompBench; realized = cascade with IVL. Best in bold, second best underlined.

$\mathcal{A}$	w_s on $\mathcal{A}$	mean	oracle	realized
$\varnothing$	–	0.4700	0.6183	0.5285
$\{0\}$	0	0.4693	0.6309	0.5277
$\{1\}$	0	**0.4755**	0.6389	0.5341
$\{2\}$	0	0.4561	0.6326	0.5279
$\{0, 1\}$	0	0.4677	<u>0.6503</u>	**0.5380**
$\{0, 1, 2\}$	0	0.4296	**0.6574**	0.5257
$\{0, 1\}$	1	<u>0.4747</u>	0.6355	<u>0.5351</u>

Table 4: **What is pruned matters.** Same pool and IVL verifier. BoN@s9 verifies all 16 at scale 9; the others cut 5 at scale 5, at random or by EAP, and verify 11. Bold: better of random and EAP.

Bench.	Pool	BoN@s9	random	EAP
T2I-CB	standard	0.5240	0.5197	**0.5285**
	DiverseVAR	0.5181	0.5119	**0.5255**
	UA (ours)	0.5372	0.5320	**0.5380**
GE2 GM	standard	0.3293	0.3115	**0.3210**
	DiverseVAR	0.3501	0.3165	**0.3366**
	UA (ours)	0.3651	0.3455	**0.3540**

Adding scale 2 buys a little oracle for a large drop in mean and scores below standard sampling, as DiverseVAR does. Keeping the text condition on the anchors ($w_s{=}1$) recovers only about half of the widening and scores lower than $w_s{=}0$.

What is pruned. Table 4 fixes the pool and verifier and changes only which candidates are cut. EAP beats a random cut of the same size on every pool, most on DiverseVAR, so it is not tuned to our sampler. It does not have to rank the pool, only to find its floor: the pruned ranks score clearly below the rest (fig. 5). Against verifying all 16 (BoN@s9), EAP never lowers the T2I-CompBench score, since a candidate removed early cannot be mis-ranked, and on GenEval2 it stays within 0.014 GM of BoN@s9 while verifying 11 candidates instead of 16, at $0.73\times$ the compute (table 15).

Operating point. We prune 30%, the best fraction on T2I-CompBench (section C.2). We run EAP at scale 5, where its worst-candidate detection saturates, and verify at scale 9: verifying the final image changes T2I-CompBench by $\leq$0.002 at 2–3$\times$ the compute (section C.1).

5.4 Other generators and prompts

Infinity-8B. The recipe and the widening transfer unchanged (table 5): UA raises the oracle by 0.020 on T2I-CompBench and 0.017 on GenEval2. Unlike at 2B, it lowers the mean slightly, yet scores higher than standard sampling in three of four selection cells and ties the fourth. The cascade stays within 0.003 of best-of-16 on T2I-CompBench and, as at 2B, gives back some GenEval2 AM, still scoring above standard best-of-16.

Switti. Switti (Voronov et al., 2024) differs from Infinity in text encoder, backbone, quantizer and scale schedule; we keep the anchor set of section 4.1 and the pruning fraction, and adapt only the pruning and verification scales to its schedule. UA raises the oracle on both benchmarks, by less than on Infinity, as Switti's standard pool is more diverse (section 3.2), and with the cascade scores higher than standard sampling on both (table 5).

Pick-a-Pic. The widening is not specific to compositional prompts: on 500 Pick-a-Pic test prompts (Kirstain et al., 2023), scored by HPSv2.1 (Wu et al., 2023) and ImageReward (Xu et al.,

Table 5: **Other generators**, N=16, IVL, same anchors and pruning fraction; GenEval2: Soft-TIFA AM. The cascade verifies scale-9 previews on Infinity-8B and the final image on Switti, where it prunes at scale 6 (2.0% of tokens). Bold: best within a model and benchmark.

Model	Bench.	Sampler	mean	oracle	best-of-16	cascade
Inf.-8B	T2I	standard	**0.5354**	0.6174	0.5497	**0.5487**
		UA	0.5314	**0.6373**	**0.5509**	0.5484
	GE2	standard	**0.7487**	0.8527	0.7923	0.7870
		UA	0.7437	**0.8699**	**0.8076**	**0.7958**
Switti	T2I	standard	**0.4608**	0.6422	0.5275	0.5286
		UA	0.4539	**0.6490**	**0.5292**	**0.5319**
	GE2	standard	**0.5539**	0.7603	**0.6741**	0.6668
		UA	0.5491	**0.7708**	0.6725	**0.6738**

Table 6: **Pick-a-Pic**, 500 prompts, N=16: per-sample mean and oracle (the best of the 16) under HPSv2.1 and ImageReward.

	HPSv2.1		ImageReward	
Sampler	mean	oracle	mean	oracle
Standard	**0.2973**	0.3142	**1.045**	1.423
UA	0.2944	**0.3182**	0.992	**1.492**

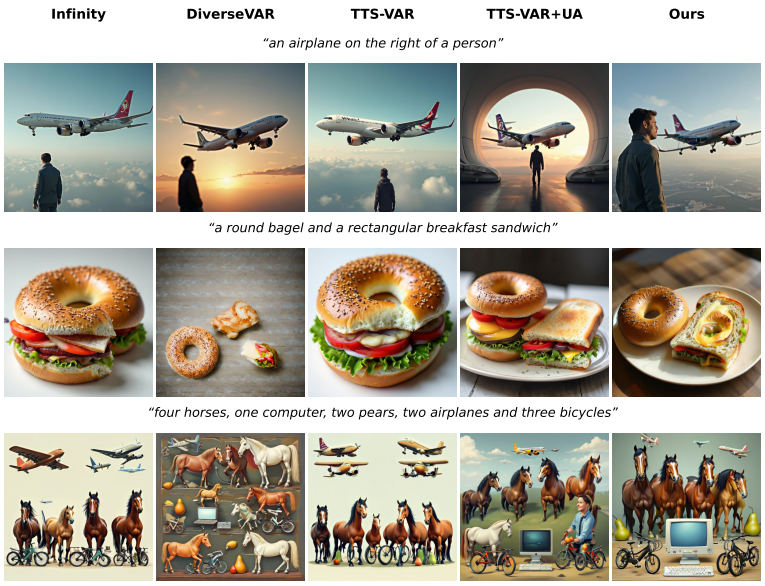


Figure 6: **Qualitative comparison.** The image each method selects for three T2I-CompBench prompts (spatial, shape, numeracy), all with N=16 candidates and the IVL verifier. Infinity and DiverseVAR are best-of-16 over standard and DiverseVAR candidates, respectively.

2023), UA raises the best-of-16 oracle under both while the per-sample mean drops slightly (table 6), and the oracle gain grows with N (section D.4).

5.5 Qualitative results

Figure 6 compares the selected images. Standard best-of-N and TTS-VAR, which both guide the earliest scales, select nearly the same composition for every prompt, consistent with the narrow early-scale distribution of section 3.2. Ours places the airplane to the right of the person and draws the bagel and the sandwich as two objects, which standard best-of-N and TTS-VAR merge into one bagel sandwich. On the counting prompt it gets the horses, the computer and the pears right.

6 Conclusion

We showed that the coarsest VAR scales, where attributes are not yet bound to objects, do not need the text condition, and built on this a test-time scaling method: unconditioned anchors widen the candidate pool at no extra cost, and early attention pruning (EAP) removes failing candidates using cross-attention the model has already computed. Together, they outperform existing diversity-boosting and test-time scaling methods at equal compute and transfer to Infinity-8B and Switti without retuning, although the gains are smaller when the standard pool is already diverse, as for Switti, and early pruning gives back some GenEval2 accuracy. More broadly, the scale at which a VAR model commits to each property of the image offers a principled guide for where to spend test-time compute.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods and analysis code, to suggest and give feedback on experiments such as ablations and diagnostic analyses, to help interpret results, and to assist with translation. We have not used generative AI tools to develop the methodology or the overall research idea, to propose or refine hypotheses, to generate synthetic data sets, or to clean and reformat data sets; theoretical models, mathematical claims and proofs, and qualitative data analysis are not applicable to this work. Additionally, we used generative AI tools to create and edit code and figures, suggest experimental parameters, draft and edit parts of the paper for readability, and propose a title. The multimodal LLM used as a verifier in our method (section 4.2) is a component of the method rather than a research aid. We have reviewed all AI-assisted work: the authors checked every reported number against the raw experimental outputs and reviewed the AI-assisted code for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, and we collect no new data. All experiments use publicly released pretrained models (Infinity, Switti, InternVL3.5, ImageReward, HPSv2.1) and public benchmarks (T2I-CompBench++, GenEval2, Pick-a-Pic), each used under its original terms. Our method is training-free and only changes how candidates are sampled and selected at inference; it adds no generation capability beyond that of the underlying models. Like any text-to-image method, it could be used to produce misleading images more efficiently, and it inherits the social biases of the generators and verifiers it relies on, which may also shape which candidate is selected. Pick-a-Pic prompts are written by real users and may contain inappropriate content; we use them only for evaluation and do not release generated images.

REPRODUCIBILITY STATEMENT

We will release our code upon publication, and all models and benchmarks we use are publicly available. Section 4 specifies the sampler (section 4.1) and the verification cascade, including the EAP signals and rule (eqs. (2) to (4)) and the verifier (section 4.2). Section 5 lists the models, benchmarks, baselines, verifiers and cost accounting, and section C justifies each operating point (pruning and verification scales, pruning fraction, attention blocks). The probing protocol is in section A, the Layout-Vendi metric and full unconditioning results in section B, and per-category results for every method in section D.1.

REFERENCES

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM transactions on Graphics (TOG)*, 42(4):1–10, 2023.

Zhekai Chen, Ruihang Chu, Yukang Chen, Shiwei Zhang, Yujie Wei, Yingya Zhang, and Xihui Liu. Tts-var: A test-time scaling framework for visual auto-regressive generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jaemin Cho, Yushi Hu, Jason Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *International conference on learning representations*, volume 2024, pp. 15625–15645, 2024.

Ho-Lam Chung, Teng-Yun Hsiao, Hsiao-Ying Huang, Chunerh Cho, Jian-Ren Lin, Zhang Ziwei, and Yun-Nung Chen. Revisiting test-time scaling: A survey and a diversity-aware method for efficient reasoning. *arXiv preprint arXiv:2506.04611*, 2025.

Do Huu Dat, Nam Hyeon-Woo, Po-Yuan Mao, and Tae-Hyun Oh. Vsc: Visual search compositional text-to-image diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19153–19162, 2025.

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, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022.

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.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Xiefan Guo, Jinlin Liu, Miaomiao Cui, Jiankai Li, Hongyu Yang, and Di Huang. Initno: Boosting text-to-image diffusion models via initial noise optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9380–9389, 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 Computer Vision and Pattern Recognition Conference*, pp. 15733–15744, 2025.

Haoran He, Jiajun Liang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, and Ling Pan. Scaling image and video generation via test-time evolutionary search. *arXiv preprint arXiv:2505.17618*, 2025.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *2023 ieee/cvf international conference on computer vision (iccv)*, pp. 20349–20360. IEEE, 2023.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 2023.

Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3563–3579, 2025.

Amita Kamath, Kai-Wei Chang, Ranjay Krishna, Luke Zettlemoyer, Yushi Hu, and Marjan Ghazvininejad. Geneval 2: Addressing benchmark drift in text-to-image evaluation. *arXiv preprint arXiv:2512.16853*, 2025.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *International Conference on Learning Representations*, volume 2025, pp. 87238–87272, 2025.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *Advances in Neural Information Processing Systems*, 37:122458–122483, 2024.

Kyungmin Lee, Xiahong Li, Qifei Wang, Junfeng He, Junjie Ke, Ming-Hsuan Yang, Irfan Essa, Jinwoo Shin, Feng Yang, and Yinxiao Li. Calibrated multi-preference optimization for aligning diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18465–18475, 2025.

Senmao Li, Kai Wang, Salman Khan, Fahad Shahbaz Khan, Jian Yang, and Yaxing Wang. Fastervar: Plug-and-play acceleration for visual autoregressive models. *arXiv preprint arXiv:2512.16483*, 2025.

Keli Liu, Zhendong Wang, Wengang Zhou, and Houqiang Li. Stepvar: Structure-texture guided pruning for visual autoregressive models. In *Proceedings of the 2026 International Conference on Multimedia Retrieval*, pp. 1626–1634, 2026.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European conference on computer vision*, pp. 423–439. Springer, 2022.

Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, et al. Inference-time scaling for diffusion models beyond scaling denoising steps. *arXiv preprint arXiv:2501.09732*, 2025.

Qianli Ma, Haotian Zhou, Tingkai Liu, Jianbo Yuan, Pengfei Liu, Yang You, and Hongxia Yang. Let's reward step by step: Step-level reward model as the navigators for reasoning. *arXiv preprint arXiv:2310.10080*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Dongmin Park, Sebin Kim, Taehong Moon, Minkyu Kim, Kangwook Lee, and Jaewoong Cho. Rare-to-frequent: Unlocking compositional generation power of diffusion models on rare concepts with llm guidance. *arXiv preprint arXiv:2410.22376*, 2024.

Mingue Park, Prin Phunyaphibarn, Phillip Y Lee, and Minhyuk Sung. Diversevar: Balancing diversity and quality of next-scale visual autoregressive models. In *The Second Workshop on Low-Level Vision Frontiers with Generative AI, Preference Optimization, Agentic Systems and World Models@ ECCV2026*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

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*, volume 2024, pp. 1862–1874, 2024.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th annual meeting of the association for computational linguistics: system demonstrations*, pp. 101–108, 2020.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pp. 8821–8831. Pmlr, 2021.

Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. *Advances in Neural Information Processing Systems*, 36:3536–3559, 2023.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann Weber. Cads: Unleashing the diversity of diffusion models through condition-annealed sampling. In *International Conference on Learning Representations*, volume 2024, pp. 23723–23755, 2024.

Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. In *International Conference on Learning Representations*, 2017.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Haotian Tang, Yecheng Wu, Shang Yang, Enze Xie, Junsong Chen, Junyu Chen, Zhuoyang Zhang, Han Cai, Yao Lu, and Song Han. Hart: Efficient visual generation with hybrid autoregressive transformer. *arXiv preprint arXiv:2410.10812*, 2024.

Weidong Tang, Xinyan Wan, Siyu Li, and Xiumei Wang. Inference-time scaling for visual autoregressive modeling by searching representative samples. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pp. 406–419. Springer, 2025.

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.

Sergey Troshin, Wafaa Mohammed, Yan Meng, Christof Monz, Antske Fokkens, and Vlad Niculae. Control the temperature: Selective sampling for diverse and high-quality llm outputs. *arXiv preprint arXiv:2510.01218*, 2025.

Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29, 2016.

Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pp. 1747–1756. PMLR, 2016.

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

Anton Voronov, Denis Kuznedelev, Mikhail Khoroshikh, Valentin Khrulkov, and Dmitry Baranchuk. Switti: Designing scale-wise transformers for text-to-image synthesis. *arXiv preprint arXiv:2412.01819*, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng Yu, Ligeng Zhu, Chengyue Wu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, et al. Sana 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. *arXiv preprint arXiv:2501.18427*, 2025.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Chenning Yu and Sicun Gao. Improving compositional generation with diffusion models using lift scores. *arXiv preprint arXiv:2505.13740*, 2025.

Qian Zhang, Xiangzi Dai, Ninghua Yang, Xiang An, Ziyong Feng, and Xingyu Ren. Var-clip: Text-to-image generator with visual auto-regressive modeling. *arXiv preprint arXiv:2408.01181*, 2024.

A LINEAR PROBING

A.1 PROTOCOL DETAILS

In this section, we clarify how the linear probing experiments is implemented.

A.1.1 PROMPTS AND IMAGES

We use the `color`, `texture` and `shape` splits of T2I-CompBench++ (Huang et al., 2023). Each prompt names exactly two attribute–object pairs ("a green bench and a red car"). The two objects are always distinct nouns, and their attributes are never equal. Images are pre-generated and filtered following the VSC pipeline (Dat et al., 2025). Each retained prompt contributes up to five images. Generators are mixed, so we report the composition on the following table.

domain	images	SD3.5	FLUX	Infinity	prompts (tr/val)	classes	chance
color	3500	2586	914	–	560 / 140	16	0.0625
texture	3500	1890	910	700	560 / 140	8	0.1250
shape	2870	1747	737	386	460 / 114	15	0.0667

Table 7: Data per domain. The train/validation split is over prompts.

A.1.2 OBJECT MASKS

Masks come from SAM3 (Carion et al., 2025), prompted with the **object noun only** and never the attribute, so a mask cannot leak the attribute it is used to predict except through layout regularities. Mask-quality statistics are computed once at the native 1024×1024 resolution and applied identically at every scale, so the sample population and class balance do not move along the scale curve. SAM3 returns an empty mask for at least one object in 6.4% (color), 15.2% (texture) and 31.7% (shape) of pairs, and these pairs are dropped.

A.1.3 REPRESENTATION EXTRACTION

We teacher-force the ground-truth VAE tokens of each image at every scale and read the hidden states from the middle transformer block, so the representation at scale s is conditioned on the true coarser scales rather than on the model's own samples.

A.1.4 MASK POOLING

To pool at a $k\times k$ token grid we downsample the full-resolution mask by *area* interpolation, giving for each cell the fraction c_i of its pixels covered by the object. Cells with $c_i \geq 0.25$ are selected and their token features averaged. If no cell reaches the threshold, we fall back to the single highest-coverage cell; we never fall back to a global mean, since that would make the binding and leakage probes identical tasks on identical features.

A.1.5 PROBES

Each probe is a single linear layer trained with Adam, learning rate 10^{-3}, batch size 256, for 50 epochs, selecting the epoch with the best validation accuracy.

For each object we train: **binding**, features pooled in object A's mask $\rightarrow$ A's attribute; and **leakage**, the same features $\rightarrow$ the *other* object's attribute. The **net signal** is binding $-$ leakage. At 1×1 the pooled feature is identical for both objects. What we report in the main table is the net signal only, and we show raw binding and leakages values in tables 8 and 9. On top of this, since attributes are not independent of object identity, we add the comparison between the one-hot encoded noun-only probe and the one with both the noun vector and features. We report the full result in table 10.

Table 8: Raw probe accuracies for Infinity-2B, null-text, middle block. *bind* is the object's own attribute, *leak* the other object's, and *net* their difference.

		color			texture			shape		
scale	grid	bind	leak	net	bind	leak	net	bind	leak	net
0	1×1	0.197	0.197	+0.000	0.189	0.189	+0.000	0.127	0.127	+0.000
1	2×2	0.278	0.263	+0.015	0.300	0.278	+0.023	0.203	0.199	+0.004
2	4×4	0.541	0.410	+0.132	0.491	0.353	+0.138	0.320	0.318	+0.003
3	6×6	0.809	0.488	+0.320	0.735	0.436	+0.299	0.471	0.419	+0.051
4	8×8	0.882	0.510	+0.372	0.841	0.438	+0.403	0.542	0.439	+0.103
5	12×12	0.900	0.510	+0.390	0.872	0.437	+0.435	0.586	0.450	+0.137
6	16×16	0.907	0.504	+0.403	0.891	0.446	+0.446	0.572	0.446	+0.127

Table 9: Raw probe accuracies for Switti, null-text, middle block. *bind* is the object's own attribute, *leak* the other object's, and *net* their difference.

		color			texture			shape		
scale	grid	bind	leak	net	bind	leak	net	bind	leak	net
0	1×1	0.197	0.197	+0.000	0.189	0.189	+0.000	0.127	0.127	+0.000
1	2×2	0.281	0.296	-0.015	0.314	0.262	+0.052	0.201	0.199	+0.001
2	3×3	0.500	0.440	+0.060	0.432	0.370	+0.063	0.254	0.254	+0.000
3	4×4	0.686	0.546	+0.140	0.565	0.452	+0.114	0.337	0.334	+0.003
4	5×5	0.790	0.610	+0.180	0.723	0.542	+0.181	0.418	0.382	+0.036
5	7×7	0.850	0.634	+0.217	0.778	0.593	+0.185	0.469	0.436	+0.033
6	9×9	0.872	0.643	+0.228	0.868	0.618	+0.250	0.556	0.468	+0.088
7	12×12	0.887	0.643	+0.245	0.900	0.646	+0.253	0.581	0.497	+0.084
8	16×16	0.893	0.638	+0.255	0.910	0.653	+0.258	0.609	0.545	+0.064

Table 10: Gain over the noun-only baseline. The two models run different scale schedules, so each model's grid is listed separately. Rows are aligned by scale index, not by resolution.

grid		Infinity-2B			Switti		
Infinity	Switti	color	texture	shape	color	texture	shape
noun-only baseline		0.307	0.734	0.449	0.307	0.734	0.449
1×1	1×1	-0.023	+0.000	+0.004	-0.013	+0.000	+0.006
2×2	2×2	+0.024	-0.011	-0.113	-0.002	-0.084	-0.146
4×4	3×3	+0.274	-0.108	-0.099	+0.213	-0.161	-0.163
6×6	4×4	+0.502	+0.023	+0.026	+0.395	-0.095	-0.063
8×8	5×5	+0.570	+0.110	+0.107	+0.500	+0.042	+0.010
12×12	7×7	+0.586	+0.156	+0.143	+0.546	+0.084	+0.068
16×16	9×9	+0.595	+0.167	+0.152	+0.574	+0.148	+0.123
–	12×12	–	–	–	+0.580	+0.181	+0.156
–	16×16	–	–	–	+0.593	+0.192	+0.182

B SCALE-WISE UNCONDITIONING: FULL RESULTS

Layout-Vendi. We measure the layout diversity of a prompt's 16 images with the Vendi score (Friedman & Dieng, 2022) of a layout kernel. Each image is resized to 224×224 and encoded by DINOv2 (ViT-L/14 with registers) (Oquab et al., 2023); the kernel entry for images i and j is the cosine similarity of their patch features at the same position, averaged over the 16×16 patch grid, so it compares where content is placed rather than what the image shows overall. The score ranges from 1 (identical layouts) to 16 (all layouts distinct) and is averaged over prompts.

Results. Table 11 lists the numbers behind fig. 2 as absolute values. On Infinity, unconditioning $\{0, 1\}$ gives the highest Layout-Vendi and oracle on both benchmarks at nearly the baseline mean,

while extending the intervention to 4×4 ($\{2\}$) lowers the mean and shrinks both gains. Switti starts from a markedly higher Layout-Vendi, and every setting widens its layouts less than on Infinity; unconditioning up to 3×3 ($\{0,1,2\}$) widens them most but also lowers the mean most, and 4×4 ($\{3\}$) loses most of the gains.

Table 11: **Full results for scale-wise unconditioning** (fig. 2). Uncond@$\mathcal{S}$ sets w_s=0 on the scales in $\mathcal{S}$; tokens: number of unconditioned tokens, the x-axis of fig. 2. N=16 per prompt; mean: average over the 16 samples; oracle: best of the 16; Layout-Vendi: Vendi score of the 16 images' DINOv2 patch-layout similarity (between 1 and 16). T2I-CompBench++: official score, 2,400 prompts; GenEval2: Soft-TIFA AM, 800 prompts. Best per model and column in bold, second best underlined.

			T2I-CompBench++			GenEval2 (AM)		
Model	Uncond@$\mathcal{S}$	tokens	Layout-Vendi	mean	oracle	Layout-Vendi	mean	oracle
Infinity-2B	none (baseline)	0	7.07	<u>0.4700</u>	0.6183	6.94	0.6192	0.7957
	$\{0\}$	1	7.89	0.4693	0.6309	7.80	<u>0.6216</u>	0.8063
	$\{1\}$	4	8.27	**0.4755**	<u>0.6389</u>	8.18	**0.6227**	<u>0.8125</u>
	$\{0,1\}$	5	**9.63**	0.4677	**0.6503**	**9.29**	0.6185	**0.8188**
	$\{2\}$	16	<u>8.33</u>	0.4561	0.6326	<u>8.41</u>	0.6051	0.8070
Switti	none (baseline)	0	10.47	**0.4608**	0.6422	10.79	**0.5539**	0.7603
	$\{0\}$	1	10.75	<u>0.4597</u>	0.6454	10.95	<u>0.5527</u>	0.7614
	$\{1\}$	4	11.20	0.4577	0.6468	11.40	0.5486	0.7627
	$\{0,1\}$	5	<u>11.49</u>	0.4539	0.6490	<u>11.55</u>	0.5491	**0.7708**
	$\{2\}$	9	11.28	0.4560	**0.6514**	<u>11.55</u>	0.5488	<u>0.7682</u>
	$\{0,1,2\}$	14	**12.32**	0.4402	<u>0.6510</u>	**12.33**	0.5360	0.7661
	$\{3\}$	16	10.91	0.4550	0.6442	11.32	0.5435	0.7566

C METHOD DETAILS

C.1 WHY PRUNE AT SCALE 5 AND VERIFY AT SCALE 9

Table 12: **Pruning scale.** UA pool, N=16; EAP (eq. (4)) with C and S read at each scale, 30% cut, IVL verifier at scale 9. Tokens: share of a candidate's tokens generated up to the pruning scale. Bottom-5: share of the 5 pruned candidates that are among the pool's 5 lowest-scoring, over prompts the rule prunes (ties split evenly; a random cut scores $5/16$=0.312). Realized: score of the selected image. GenEval2: Soft-TIFA GM. Best per column among the pruning scales in bold, second best underlined.

		T2I-CompBench		GenEval2	
Prune at	Tokens	bottom-5	realized	bottom-5	realized
no pruning (BoN@s9)	–	–	0.5372	–	0.3651
scale 3	0.5%	0.389	0.5350	0.386	0.3578
scale 4	1.2%	0.396	<u>0.5371</u>	0.415	0.3558
scale 5 (ours)	2.5%	<u>0.399</u>	**0.5378**	<u>0.435</u>	0.3540
scale 6	5.0%	**0.401**	0.5368	**0.440**	<u>0.3643</u>
scale 7	8.8%	0.399	0.5356	<u>0.435</u>	**0.3663**

Pruning scale. EAP only has to find the worst candidates, and this ability saturates at scale 5 (table 12): reading the signal earlier finds significantly fewer of them (paired test over prompts, $p<0.05$), while reading it later, at 2–3.5× the tokens, finds no more. We therefore prune at scale 5, the cheapest scale at which the signal is saturated. On T2I-CompBench, where we set all cascade hyperparameters, scale 5 also achieves the highest score. On GenEval2, pruning later scores higher although it finds the bottom no better, because it cuts the pool's best candidate less often; there, scale 7 is the better setting if the extra tokens are affordable.

Verification scale. Verifying the finished images instead of the scale-9 previews costs 2–3× the compute (tables 15 and 16). It leaves T2I-CompBench essentially unchanged and raises GenEval2 only for the DiverseVAR and UA pools.

C.2 WHY PRUNE 30%

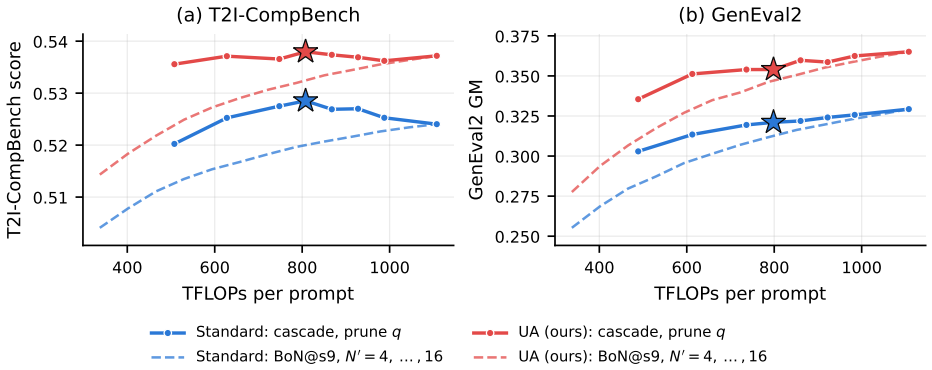


Figure 7: **Pruning fraction.** Cascade with the IVL verifier as the pruned fraction q goes from 0 (BoN@s9, right end) to 60%; stars mark q=30%. Dashed: BoN@s9 over a random subset of N' candidates, the same compute without the signal. TFLOPs as in table 2.

Figure 7 sweeps the fraction pruned at scale 5. We fix q=30%, where the T2I-CompBench score peaks. On GenEval2 this cut gives back a little GM relative to BoN@s9, far less than larger cuts, and still beats a random cut of the same compute; best-of-N on finished images at the same compute falls further behind (figs. 4 and 8). On Switti (IVL), where q was not re-tuned, every fraction from 10% to 40% scores at or above best-of-16 on T2I-CompBench; on GenEval2 AM the UA pool is insensitive to it, while the standard pool gives back a little more the more is pruned.

C.3 WHICH TRANSFORMER BLOCKS

Table 13: **Where the EAP signal lives.** T2I-CompBench, EAP (eq. (4)) with both C and S computed from the scale-5 cross-attention of one group of Infinity-2B's 32 blocks, 30% cut, IVL verifier at scale 9. Survival: share of the 2,327 prompts with a signal whose best candidate survives the cut; realized: score of the selected image over all 2,400 prompts. Random cut: expectation over 200 draws. Best per column in bold.

	Standard pool		UA pool	
Blocks	survival	realized	survival	realized
early (0–10)	**0.744**	0.5281	**0.737**	0.5372
mid (11–21)	0.737	0.5253	0.714	0.5354
late (22–31)	0.682	0.5158	0.693	0.5336
all (0–31)	**0.744**	0.5274	0.734	0.5339
early + mid (ours)	**0.744**	**0.5285**	**0.737**	**0.5380**
random cut	0.687	0.5193	0.689	0.5321

The late blocks carry no EAP signal: cutting with them is no better than a random cut (table 13). The early and middle blocks both carry it; averaging the two, a choice fixed in a preliminary study on GenEval2, achieves the highest score in both pools, and adding the late blocks lowers it.

D ADDITIONAL EXPERIMENTAL RESULTS

D.1 FULL T2I-COMPBENCH AND GENEVAL2 TABLES

Table 14: **Pool quality by subtask**, $N{=}16$: mean over the 16 candidates and oracle (the best of them), the per-subtask counterpart of table 1; $\{0, 1, 2\}$ is the ablation of table 3. T2I-CompBench: official score per category, overall = mean of the eight. GenEval2: Soft-TIFA atom accuracy per skill; the oracle is the candidate with the highest prompt AM, and AM is the prompt-level score. Best per column among the mean rows and among the oracle rows in bold, second best underlined.

Sampler		color	shape	texture	spatial	3D-sp.	non-sp.	numer.	complex	overall
Standard	mean	**0.746**	0.486	**0.625**	<u>0.236</u>	**0.407**	**0.310**	<u>0.567</u>	**0.382**	**0.470**
DiverseVAR	mean	0.608	0.463	0.566	0.181	0.324	0.302	0.492	0.320	0.407
UA $\{0, 1\}$	mean	<u>0.719</u>	**0.499**	**0.625**	**0.251**	<u>0.395</u>	<u>0.309</u>	**0.568**	<u>0.374</u>	<u>0.468</u>
UA $\{0, 1, 2\}$	mean	0.653	<u>0.487</u>	<u>0.597</u>	0.204	0.342	0.301	0.517	0.335	0.430
Standard	oracle	0.884	0.699	0.808	0.409	0.574	0.325	0.801	0.447	0.618
DiverseVAR	oracle	**0.917**	**0.766**	<u>0.845</u>	<u>0.514</u>	0.601	**0.333**	0.813	0.454	<u>0.656</u>
UA $\{0, 1\}$	oracle	0.907	0.740	0.830	0.484	**0.617**	<u>0.328</u>	**0.837**	**0.458**	0.650
UA $\{0, 1, 2\}$	oracle	<u>0.916</u>	<u>0.759</u>	**0.846**	**0.523**	<u>0.605</u>	<u>0.328</u>	<u>0.825</u>	<u>0.457</u>	**0.657**

Sampler		object	count	attribute	position	verb	AM
Standard	mean	**0.865**	**0.438**	<u>0.597</u>	<u>0.344</u>	**0.291**	**0.619**
DiverseVAR	mean	0.786	0.353	0.567	0.271	0.191	0.547
UA $\{0, 1\}$	mean	<u>0.861</u>	<u>0.429</u>	**0.603**	**0.353**	<u>0.285</u>	<u>0.618</u>
UA $\{0, 1, 2\}$	mean	0.855	0.381	0.583	0.324	0.251	0.589
Standard	oracle	0.969	<u>0.707</u>	0.731	0.549	0.463	0.796
DiverseVAR	oracle	0.967	0.683	**0.794**	0.582	<u>0.469</u>	0.805
UA $\{0, 1\}$	oracle	<u>0.973</u>	**0.722**	0.758	**0.631**	**0.483**	**0.819**
UA $\{0, 1, 2\}$	oracle	**0.979**	<u>0.707</u>	<u>0.776</u>	<u>0.609</u>	0.445	<u>0.816</u>

Table 14 breaks the pool quality of table 1 down by subtask. UA keeps the mean of standard sampling in nearly every subtask, color being the exception, and raises the oracle most on spatial, 3D-spatial, shape and numeracy prompts and on position atoms, the subtasks that depend on layout. DiverseVAR reaches a higher oracle on color, shape, texture and spatial prompts and on attribute atoms, but lowers the mean in every subtask, and by more than UA in each; with a verifier, its pool scores lower than UA's overall on both benchmarks (table 2). Extending the anchors to $\{0, 1, 2\}$ matches DiverseVAR's oracle in most subtasks while keeping a higher mean in nearly all of them: at an equal oracle, anchoring on the model's own prior costs less mean than noising the text condition.

Table 15: **T2I-CompBench by category**, scored on the selected image, N=16 except the single sample. Best-of-N verifies the finished images as in table 2; BoN@s9 verifies the scale-9 previews and finishes only the winner. Cost per prompt (wall-clock in s, TFLOPs) as in table 2, lower is better. Best per column within a verifier block in bold, second best underlined; non-spatial, a CLIP score on which every method lies within 0.004, is left unmarked.

Sampler	Selection	color	shape	texture	spatial	3D-sp.	non-sp.	numer.	complex	overall	wall	TFLOPs
Standard	single sample	0.746	0.486	0.625	0.236	0.407	0.310	0.567	0.382	0.470	2.3	102
Verifier: IR												
Standard	best-of-N	0.799	0.580	0.726	0.253	0.439	0.313	0.627	**0.402**	0.517	22.5	1,643
Standard	BoN@s9	0.796	0.587	0.725	0.251	0.437	0.312	0.624	0.398	0.516	9.9	597
DiverseVAR	best-of-N	0.796	**0.638**	**0.745**	0.278	0.413	0.312	0.623	0.385	0.524	32.4	1,777
DiverseVAR	BoN@s9	0.797	0.632	0.738	0.275	0.416	0.311	0.631	0.384	0.523	22.5	986
Standard	TTS-VAR	0.773	0.571	0.686	0.246	0.423	0.312	0.617	0.398	0.503	34.8	1,369
UA	TTS-VAR	0.797	0.597	0.720	**0.294**	0.422	0.310	0.608	0.390	0.517	34.7	1,369
UA	best-of-N	**0.819**	0.616	0.735	0.276	0.438	0.312	0.636	**0.402**	0.529	22.5	1,643
UA	BoN@s9	0.816	0.613	0.731	0.271	0.434	0.311	0.646	0.400	0.528	10.1	597
UA	cascade	0.814	0.613	0.728	0.288	**0.446**	0.311	**0.647**	0.401	**0.531**	**7.0**	**452**
Verifier: IVL												
Standard	best-of-N	0.805	0.576	0.733	0.323	0.412	0.311	0.642	0.401	0.525	32.5	2,154
Standard	BoN@s9	0.809	0.571	0.731	0.318	0.411	0.312	0.637	**0.403**	0.524	17.8	1,108
DiverseVAR	best-of-N	0.798	0.614	0.736	0.346	0.338	0.309	0.621	0.381	0.518	40.9	2,288
DiverseVAR	BoN@s9	0.803	**0.615**	**0.737**	0.351	0.322	0.310	0.625	0.383	0.518	31.2	1,497
Standard	TTS-VAR	0.783	0.533	0.689	0.286	0.405	0.311	0.627	0.388	0.503	43.0	1,816
UA	TTS-VAR	0.792	0.563	0.701	0.321	0.379	0.311	0.633	0.393	0.512	42.6	1,816
UA	best-of-N	**0.819**	0.596	0.729	0.364	0.390	0.312	0.668	**0.403**	0.535	33.3	2,154
UA	BoN@s9	0.813	0.602	0.730	0.369	0.397	0.312	**0.672**	0.402	0.537	17.8	1,108
UA	cascade	0.815	0.594	0.735	**0.371**	**0.415**	0.312	0.662	0.401	**0.538**	**12.7**	**808**

Table 16: **GenEval2 by skill**, Soft-TIFA atom accuracy per skill, AM and GM, IVL verifier; best-of-N, BoN@s9 and cost as in table 15 (cost measured on T2I-CompBench). Best per column among the N=16 rows in bold (lowest for cost), second best underlined.

Sampler	Selection	object	count	attribute	position	verb	AM	GM	wall	TFLOPs
Standard	single sample	0.865	0.438	0.597	0.344	0.291	0.6192	0.1614	2.3	102
DiverseVAR	single sample	0.786	0.353	0.567	0.271	0.191	0.5467	0.1112	2.7	111
UA	single sample	0.861	0.429	0.603	0.353	0.285	0.6185	0.1622	2.3	102
Standard	best-of-N	0.922	0.592	0.656	0.435	0.297	0.7141	0.3297	32.5	2,154
Standard	BoN@s9	0.927	0.593	0.650	0.426	0.324	0.7129	0.3293	17.8	1,108
DiverseVAR	best-of-N	0.906	0.594	0.687	0.460	0.329	0.7223	0.3608	40.9	2,288
DiverseVAR	BoN@s9	0.899	0.588	**0.696**	0.441	0.300	0.7148	0.3501	31.2	1,497
Standard	TTS-VAR	0.898	0.544	0.635	0.402	0.307	0.6807	0.2720	43.0	1,816
UA	TTS-VAR	0.896	0.558	0.630	0.411	0.356	0.6893	0.2848	42.6	1,816
UA	best-of-N	0.928	**0.624**	0.684	**0.515**	0.331	**0.7439**	**0.3793**	33.3	2,154
UA	BoN@s9	0.925	0.612	0.690	0.510	0.359	0.7380	0.3651	17.8	1,108
UA	cascade	**0.931**	0.610	0.685	0.499	**0.375**	0.7370	0.3540	**12.7**	**808**

On T2I-CompBench the cascade's gain is broad across categories, largest on spatial and 3D-spatial, while DiverseVAR leads on shape and texture (table 15). On GenEval2 its gain over standard best-of-N is largest on position and verb atoms (table 16).

D.2 Quality against compute on GenEval2

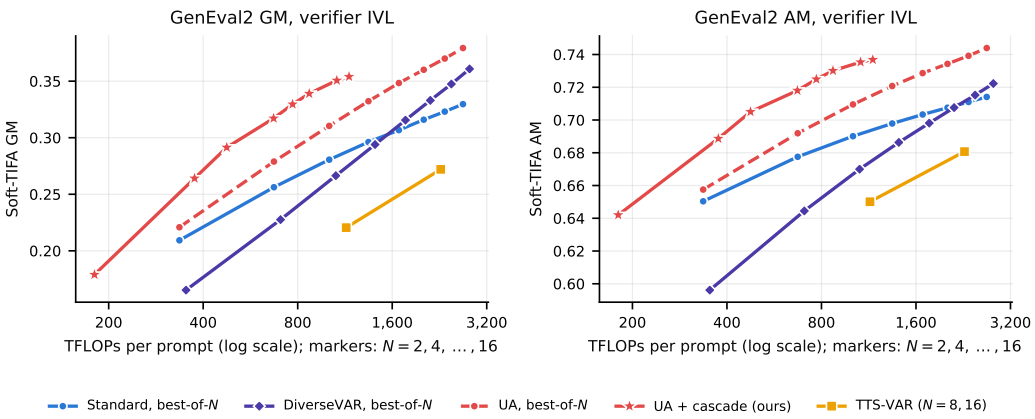


Figure 8: **Quality against compute**, GenEval2, IVL, constructed as fig. 4. Cost counts GenEval2's verifier questions (7.68 per prompt against 3.77 on T2I-CompBench), so every point that verifies sits further right than in table 2; the cascade at $N{=}16$ costs 1,166 TFLOPs instead of 808.

At $N{=}16$ the cascade gives back GM against UA best-of-16 (table 2), but not at equal compute: at the cascade's cost it scores higher than every best-of-N and TTS-VAR on both GM and AM, and UA best-of-N needs substantially more compute to match it (fig. 8). The only baseline that reaches a higher GM, DiverseVAR best-of-16, uses over $2\times$ the compute and scores a lower AM.

D.3 Random codebook tokens

Table 17: **Two ways to widen the anchors.** T2I-CompBench, $N{=}16$. Random codebook tokens: the anchor scales take uniformly random codebook indices; guidance off: $w_s{=}0$ on the same scales (UA). Both arms of a row share one random stream, so the $\{1\}$ row differs slightly from table 3. Standard sampling: mean 0.470, oracle 0.618.

	random codebook tokens		guidance off (UA)	
$\mathcal{A}$	mean	oracle	mean	oracle
$\{0\}$	0.469	0.628	0.469	**0.631**
$\{1\}$	0.475	0.634	0.476	**0.635**
$\{0,1\}$	0.469	0.645	0.468	**0.650**

Both widen the pool at an unchanged mean, and guidance off reaches the higher oracle at every anchor set, significantly so for $\{0,1\}$ (table 17). Guidance off also needs no injected codes: it changes only the inference schedule.

D.4 Pick-a-Pic

Table 18: **Pick-a-Pic**, 500 prompts, $N{=}16$ candidates per prompt: per-sample mean and best-of-N oracle under HPSv2.1 and ImageReward; $N{=}4$ and 8 are the expected best of N candidates drawn from the 16.

	HPSv2.1				ImageReward			
Sampler	mean	$N{=}4$	$N{=}8$	$N{=}16$	mean	$N{=}4$	$N{=}8$	$N{=}16$
Standard	**0.2973**	0.3074	0.3111	0.3142	**1.045**	1.288	1.362	1.423
UA	0.2944	**0.3090**	**0.3140**	**0.3182**	0.992	**1.325**	**1.418**	**1.492**

UA lowers the per-sample preference score slightly but raises the best-of-N oracle under both scorers at every N, by more as N grows (table 18); both the drop in mean and the oracle gains are significant (paired t-test over prompts, $p<0.01$).