ATHENA: TEST-TIME STEERING FOR COUNT FIDELITY IN TEXT-TO-IMAGE DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image diffusion models achieve high visual fidelity yet exhibit systematic failures when prompts specify explicit object counts. We introduce ATHENA, a model-agnostic, training-free framework for improving count fidelity through test-time steering. ATHENA constructs a cardinality-specific contrastive steering signal by varying the numerical condition in the input prompt, enabling targeted count intervention during denoising while preserving the underlying semantic content. However, different generation trajectories develop different count errors during denoising, making a fixed steering signal insufficient. To address this, we further incorporate intermediate count feedback to identify emerging cardinality errors, construct trajectory-specific corrections, and adjust the steering strength based on the observed response. Across multiple text-to-image diffusion backbones and counting benchmarks, we show that ATHENA consistently improves count fidelity over unsteered generation and existing counting methods, with particularly strong gains in challenging counting settings while maintaining favorable accuracy–runtime trade-offs. Our code and data are publicly available at `https://anonymous.4open.science/r/ATHENA-8527`.

1 INTRODUCTION

Text-to-image (T2I) diffusion models have achieved remarkable progress in high-quality image generation, enabling the synthesis of visually realistic and semantically aligned images from natural language prompts (Rombach et al., 2022; Saharia et al., 2022).



Figure 1: Count fidelity failures across T2I generative models. Despite explicit numerical specifications in the prompts, generated images can contain incorrect object counts across different generative backbones, even for simple objects and scenes.

Despite their strong generative capabilities, T2I models continue to exhibit systematic failures in numerical control when prompts specify explicit object counts (Kajić et al., 2024; Binyamin et al., 2025). Even for visually simple and well-defined categories, prompts such as *"a photo of eight cows in a field"* or *"ten apples on a table"* can produce images with missing instances, duplicated objects, or ambiguous merges that obscure instance boundaries. As illustrated in Figure 1, these counting errors occur across different generative backbones, indicating that reliable object-count control remains a broader challenge in T2I generation rather than a limitation of a particular model.

From a modeling perspective, object counting is particularly challenging because the number and spatial organization of object instances emerge progressively during the generation trajectory. Prior analyses of diffusion models suggest that coarse scene composition is established early in the denoising process, while later stages increasingly refine appearance and local details (Tinaz et al.,

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.

2025; Lee et al., 2025). For object counting specifically, this means that cardinality errors can emerge early and become increasingly difficult to correct as the scene structure is established. Effective count control should therefore act while the scene structure remains sufficiently malleable.

Intervening during generation raises a key challenge: how can the numerical constraint be targeted without unnecessarily altering the remaining scene semantics? A natural way to construct such an intervention is to contrast prompts that describe the same scene while differing in their numerical specification. By preserving the shared semantic context and varying the requested cardinality, this contrast provides a count-specific signal for steering the generation trajectory.

In this work, we introduce ATHENA, a model-agnostic, training-free framework for improving count fidelity through test-time steering. ATHENA uses cardinality-specific steering to enable targeted count intervention during generation without retraining or architectural modifications. The framework can further incorporate intermediate count feedback to make the intervention responsive to the cardinality emerging during generation, and adapt the steering strength based on the observed response.

Contributions. Our main contributions are as follows:

- We propose ATHENA, a **model-agnostic**, **training-free** framework for improving object-count fidelity in T2I diffusion and flow-matching models through cardinality-specific test-time steering. ATHENA requires no architectural changes or retraining and can be applied through lightweight test-time interventions.
- We introduce the ATHENA-COUNT dataset, a new evaluation benchmark that complements existing counting datasets with **challenging object categories** (e.g., accordions), **compositional contexts** (e.g., next to a river), **distractor objects** (e.g., with a person), and **relational constraints** (e.g., arranged in a circle).
- Through extensive experiments across multiple T2I backbones and counting benchmarks, we demonstrate substantial improvements in count fidelity over unsteered generation and existing counting methods, with gains of up to **22 percentage points** in exact-count accuracy, up to $2.5\times$ faster image generation, and approximately $4\times$ lower memory usage relative to evaluated prior baselines.
- We further assess the robustness and generalizability of ATHENA beyond the primary evaluation setting, demonstrating consistent improvements under **alternative evaluation protocols** and **multi-object counting scenarios**.

2 RELATED WORK

Accurate object-count control remains a persistent challenge in T2I generation, motivating several approaches for explicit cardinality control. *Make It Count* (Binyamin et al., 2025) corrects under-generation through learned modification of internal object layouts, while *CountCluster* (Lee et al., 2025) encourages the requested number of object clusters through early cross-attention optimization. Other approaches introduce explicit counting signals during inference: *Counting Guidance* (Kang et al., 2025) and *YOLO-Count* (Zeng et al., 2025) update embeddings using gradients of specialized counting models, whereas *CountDiffusion* (Li et al., 2025) uses intermediate count estimates to guide attention-based correction. *Detection-Driven Object Count Optimization* (Zafar et al., 2026) instead optimizes a counting-token embedding using counting and detection signals from generated images. More recently, *CountLoop* (Mondal et al., 2026) combines structured layout generation with multimodal critique and iterative refinement for training-free high-instance count control.

Interventions during generation can take different forms depending on how the target cardinality and the evolving generation state are used. *CountCluster* (Lee et al., 2025) imposes the target count by organizing early cross-attention into a prescribed number of spatial clusters, while *CountDiffusion* (Li et al., 2025) estimates the emerging count from an intermediate prediction and uses it to construct an attention-based correction. In contrast, ATHENA constructs a steering direction by comparing the model predictions under the original prompt and a modified version that differs only in its numerical specification. Intermediate count estimation is optionally used to determine this modification based on the count emerging during generation.

More broadly, inference-time guidance provides a general mechanism for controlling diffusion generation without modifying model parameters. Classifier-free guidance strengthens prompt conditioning by combining conditional and unconditional model predictions (Ho & Salimans, 2022), while contrastive guidance uses minimally differing prompts to steer specific semantic factors (Wu & De la Torre, 2024). Attention-based approaches provide an alternative form of test-time control through cross-attention manipulation (Hertz et al., 2023; Chefer et al., 2023). ATHENA instead specializes inference-time steering for object counting by modifying the numerical specification of the prompt while preserving its remaining semantic content.

3 METHOD

3.1 PROBLEM SETUP

We study test-time control of object count fidelity in pretrained T2I generative models. Given a text prompt p specifying a target count k for a particular object category, the generation process produces a sequence of latent states $\{z_t\}_{t=0}^{T}$, where z_T denotes the initial noisy latent and z_0 the final clean latent. Our objective is to improve the accuracy with which the final generated image satisfies the specified cardinality. We assume access to intermediate latent states and model predictions during sampling, and all interventions are performed at inference time without retraining or modifying the generator parameters. Further details on diffusion sampling and notation are provided in Section A.

3.2 CARDINALITY-SPECIFIC STEERING FRAMEWORK

ATHENA is a training-free framework for improving object count fidelity through targeted intervention during the generation process. It operates directly on the sampling trajectory to steer generation toward the specified cardinality. This formulation underlies the control strategies introduced later in section 3.4. We next describe the core steering mechanism and its geometric motivation.

3.2.1 STEERING MECHANISM

At each sampling step t, ATHENA evaluates the denoiser twice on the current latent state z_t: once with the original prompt p, producing $\epsilon_t \triangleq \epsilon_\theta(t, z_t, p)$, and once with a *control prompt* $\hat{p}$, obtained by modifying the numerical specification of p, producing $\hat{\epsilon}_t \triangleq \epsilon_\theta(t, z_t, \hat{p})$ (see Figure 2). The two predictions are combined to form the steered noise estimate

$$\tilde{\epsilon}_t = \epsilon_t + \gamma(\epsilon_t - \hat{\epsilon}_t), \tag{1}$$

where $\gamma \geq 0$ controls the steering strength. The sampler update (Equation (6)) is then applied using $\tilde{\epsilon}_t$ in place of ϵ_t to obtain the next latent state z_{t-1}. Before the sampler update, we normalize $\tilde{\epsilon}_t$ to match the norm of ϵ_t, preserving the scale of the original denoiser prediction while modifying its direction through the steering term.

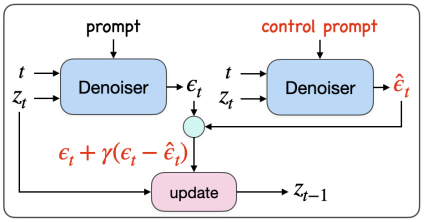


Figure 2: ATHENA steering block. The denoiser is evaluated with the original and control prompts to obtain ϵ_t and $\hat{\epsilon}_t$, which are combined to obtain the latent state z_{t-1} for test-time count control.

The difference term $(\epsilon_t - \hat{\epsilon}_t)$ captures the change in the denoiser prediction induced by replacing the original prompt p with the control prompt $\hat{p}$. We construct $\hat{p}$ by modifying only the object-count specification in p while preserving the remaining semantic content. This construction provides a cardinality-specific contrast between the two conditioning signals, which ATHENA uses to steer the sampling trajectory toward the target count. The steering signal is computed entirely through forward denoiser evaluations and requires no backpropagation.

We consider two forms of control prompts: a *count-agnostic* control prompt, which removes the explicit cardinality specification from p and provides conditioning without an explicit target count, and a *feedback-based* control prompt, which replaces the target count with an estimated count obtained from an intermediate decoded generation. These two constructions provide different control conditions for the same steering mechanism and form the basis of the ATHENA strategies described in Section 3.4.

Table 1: Intermediate-to-final count agreement across T2I backbones. G-DINO counts at step 20 for FLUX.1-dev and SD 3.5 Large and step 30 for SDXL are compared with the corresponding final counts after 50 sampling steps.

FLUX.1-dev			**SD 3.5 Large**			**SDXL**		
Exact Match (%) ↑	MAE ↓	MSE ↓	Exact Match (%) ↑	MAE ↓	MSE ↓	Exact Match (%) ↑	MAE ↓	MSE ↓
72.67	0.453	1.025	76.40	0.509	2.323	70.19	0.584	2.298

3.2.2 Geometric Interpretation

For a conditioning prompt q, let $\epsilon_{t,q}$ denote the denoiser output. Under the standard noise-prediction objective (Ho et al., 2020), $\epsilon_{t,q} \approx \mathbb{E}[\xi \mid z_t, q]$, where $\xi \sim \mathcal{N}(0, I)$ is the injected noise and σ_t is its scale. Defining the conditional score as $s_{t,q}(z_t) \triangleq \nabla_{z_t} \log p_t(z_t \mid q)$, Gaussian corruption gives:

$$\mathbb{E}[\xi \mid z_t, q] = -\sigma_t s_{t,q}(z_t) \Rightarrow -\frac{1}{\sigma_t}\left(\epsilon_{t,p} - \epsilon_{t,\hat{p}}\right) \approx s_{t,p}(z_t) - s_{t,\hat{p}}(z_t). \tag{2}$$

Thus, up to a known scale and sign, the contrast between the denoiser outputs approximates the difference between their conditional scores.

Let $\mathcal{M}_{t,q}$ denote the local high-probability-density manifold of latent states under $p_t(z_t \mid q)$. Prior work shows that, under Gaussian corruption and low-noise manifold assumptions, the score points toward the data manifold and can approximate the local projection onto it (Alain & Bengio, 2014; Stanczuk et al., 2024; Kharitenko et al., 2026). Motivated by these results, we assume that the conditional density is locally isotropic around $\mathcal{M}_{t,q}$, yielding

$$s_{t,q}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,q}}(z_t) - z_t}{\tau_t^2}, \qquad s_{t,p}(z_t) - s_{t,\hat{p}}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,p}}(z_t) - \Pi_{\mathcal{M}_{t,\hat{p}}}(z_t)}{\tau_t^2}, \tag{3}$$

where $\Pi_{\mathcal{M}_{t,q}}$ denotes local projection onto $\mathcal{M}_{t,q}$ and τ_t^2 is the local variance. If $\hat{p}$ is a *count-agnostic prompt*, $\mathcal{M}_{t,\hat{p}}$ can be interpreted as a count-marginalized region containing modes with different object counts. Alternatively, if $\hat{p}$ specifies the actual object count of the emerging image represented by z_t, then $\mathcal{M}_{t,\hat{p}}$ describes the count-specific region containing the *current trajectory*. In both cases, we assume that z_t is aligned with the remaining semantics of p, which is reasonable for state-of-the-art text-to-image models. Under this assumption, the control condition locally describes z_t, such that $\Pi_{\mathcal{M}_{t,\hat{p}}}(z_t) \approx z_t$. Hence,

$$s_{t,p}(z_t) - s_{t,\hat{p}}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,p}}(z_t) - z_t}{\tau_t^2}. \tag{4}$$

Combining Equation (2) and Equation (4) yields

$$\epsilon_{t,p} - \epsilon_{t,\hat{p}} \approx -\frac{\sigma_t}{\tau_t^2}\left(\Pi_{\mathcal{M}_{t,p}}(z_t) - z_t\right). \tag{5}$$

Because the predicted clean latent depends on $-\epsilon_t$, adding this contrast approximately shifts its clean estimate toward the target-conditioned manifold.

We provide a more detailed derivation and discussion of these assumptions in Section B. Together, these results show that ATHENA's contrastive update approximately shifts the predicted clean latent toward the target-count manifold, providing a geometric justification for its steering mechanism.

3.3 Feedback-Informed Cardinality Control

The steering mechanism in Section 3.2.1 defines how to intervene given a control prompt $\hat{p}$. To make this control responsive to the cardinality emerging during generation, we introduce feedback-based control, which uses an intermediate estimate of the object count to inform the control prompt.

For feedback-based control, ATHENA estimates the cardinality emerging during generation from a clean-image estimate at an intermediate sampling step t_{est}. We decode the corresponding clean-latent estimate $\hat{z}_0^{(t_{\text{est}})}$ into an image and apply an object detector to obtain an estimated count $\hat{c}$. Very early reconstructions contain insufficient object structure for meaningful count estimation, whereas waiting until later sampling steps increases the computational cost of obtaining the estimate. We therefore estimate the emerging count at an intermediate stage to balance estimation reliability and cost.

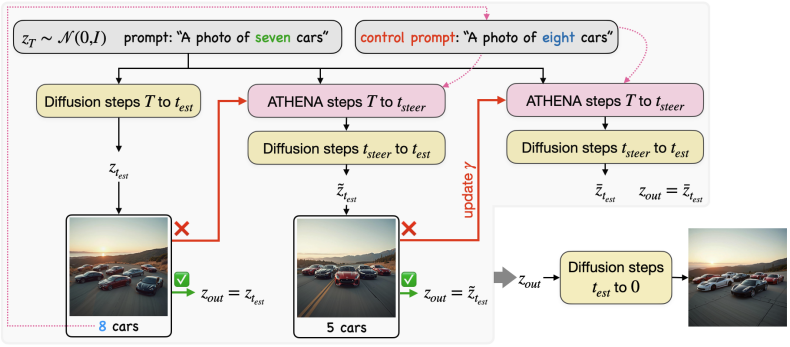


Figure 3: ATHENA-Adaptive pipeline. The method estimates object count at an intermediate diffusion step, applies early-stage prompt-based steering, adaptively adjusts the steering strength once based on the observed error direction, and completes generation using standard diffusion.

Prior analyses suggest that coarse scene structure and object organization emerge during earlier stages of generation (Tinaz et al., 2025; Lee et al., 2025). To assess whether this intermediate structure is informative of the eventual cardinality, we evaluate intermediate-to-final count agreement across three T2I backbones using CoCoCount prompts (Binyamin et al., 2025) and Grounding DINO (G-DINO) (Liu et al., 2024) as the object detector. As shown in Table 1, the intermediate and final detected counts agree exactly in 70.19–76.40% of generations, with MAE ranging from 0.453 to 0.584. This analysis measures intermediate-to-final count agreement rather than the absolute counting accuracy of G-DINO; further details and an illustration are provided in Section C.

The resulting count estimate informs the control condition but does not directly define the correction direction. The steering signal remains determined by the contrast between the denoiser predictions under the original and control prompts in Equation (1). The specific use of this feedback in the ATHENA control strategies is described next.

3.4 ATHENA CONTROL STRATEGIES

We consider three ATHENA control strategies that share the same test-time steering mechanism but differ in how the control prompt and steering strength are determined. ATHENA-Static uses a fixed count-agnostic control prompt, ATHENA-Feedback uses intermediate count feedback to construct a feedback-informed control prompt, and ATHENA-Adaptive additionally adjusts the steering strength based on the observed behavior under steering.

ATHENA-Static. ATHENA-Static uses a fixed, count-agnostic control prompt $\hat{p}$, constructed by removing the explicit cardinality from the original prompt p while preserving the remaining semantic content (e.g., "a photo of seven apples" $\rightarrow$ "a photo of apples"). No intermediate count estimation or feedback is used. Starting from the initial sampling step, the steering mechanism in Equation (1) is applied until a cutoff t_{steer}, while the scene structure remains sufficiently malleable, after which sampling proceeds without steering. By avoiding intermediate count estimation and feedback, this strategy incurs the lowest computational overhead among the three strategies. Further details are provided in Section D.1.

ATHENA-Feedback. Unlike ATHENA-Static, which uses a fixed count-agnostic control prompt, ATHENA-Feedback uses a single intermediate count estimate to construct a control prompt informed by the cardinality emerging during generation. Sampling first proceeds without steering until t_{est}, where the intermediate count $\hat{c}$ is estimated. If $\hat{c} = k$, where k denotes the target count, sampling continues without count correction. Otherwise, generation is restarted from the same initial noise, and the control prompt $\hat{p}$ is constructed by replacing the target count k in p with the estimated emerging count $\hat{c}$, while preserving the remaining semantic content. Starting from the initial sampling step of the restarted trajectory, the steering mechanism in Equation (1) is applied until t_{steer}, after which sampling proceeds without steering. No additional count estimates or steering-strength adjustments are performed. Further details are provided in Section D.2.

ATHENA-Adaptive. While ATHENA-Feedback adapts the control prompt to the estimated emerging count, the magnitude of the resulting correction is governed by the steering strength γ. If γ is too

small, steering may move the generation toward the target without fully correcting the count; if γ is too large, the correction may overshoot the target and reverse the error direction. ATHENA-Adaptive uses the observed response to an initial steering intervention to adjust γ accordingly.

As illustrated in Figure 3, the strategy first follows the feedback procedure to estimate the baseline count $\hat{c}$ at t_{est} without steering and constructs the feedback-informed control prompt when correction is needed. A steered generation is then performed using an initial steering strength γ. At the same t_{est}, the resulting count is estimated again. If it reaches the target k, no further adjustment is made. Otherwise, γ is adjusted according to the direction of the remaining count error: if the error remains on the same side of the target as the baseline error, γ is doubled; if the error crosses the target, γ is halved. A final steered generation is then performed using the adjusted steering strength.

This adjustment is deliberately bounded to a single update of γ, rather than iteratively tuning the steering strength until the target count is reached. Consequently, ATHENA-Adaptive requires at most two steered generations after the initial unsteered count estimate and introduces no additional learned components. Pseudocode is provided in Section D.3.

4 EXPERIMENTS

We evaluate ATHENA across multiple T2I generative backbones and counting benchmarks to assess its effectiveness, computational efficiency, and generalization. Our experiments compare the proposed control strategies with existing count-control methods across counting tasks of varying complexity. We further examine whether the observed improvements persist under independent evaluation and extend beyond the primary experimental setting to recent T2I models, external benchmarks, and simultaneous multi-object counting constraints.

4.1 EXPERIMENTAL SETUP

Models. We conduct our primary experiments on three T2I generative backbones: SDXL (Podell et al., 2023), SD 3.5 Large (Esser et al., 2024), and FLUX.1-dev (Labs, 2024; Labs et al., 2025). We additionally evaluate ATHENA on the recent FLUX.2-dev model (Labs, 2025) in our generalization experiments, with Qwen-Image (Wu et al., 2025) included as an unsteered reference.

Datasets. Our primary evaluation uses three counting benchmarks. *CoCoCount* (Binyamin et al., 2025) consists of simple prompts with explicit numerical constraints and minimal scene context. To enable systematic evaluation across target cardinalities, we construct *CoCoCount-E*, a balanced extension spanning target counts from 2 to 10 with consistent prompt structure. We further introduce the ATHENA-COUNT dataset, comprising 360 prompts with target counts from 2 to 10 and organized into four levels of increasing complexity: challenging object categories, scene context, distractor objects, and relational constraints. For multi-object evaluation, we additionally construct *CoCoCount-MO*, comprising 200 prompts formed by sampling semantically compatible pairs of object categories and their associated target counts from CoCoCount, enabling evaluation under multi-object cardinality constraints. Additional construction details, examples, and dataset statistics are provided in Section E.

Baselines. We compare ATHENA against unsteered generation and three representative count-control methods: CountGen (Binyamin et al., 2025), Counting Guidance (Kang et al., 2025), and YOLO-Count (Zeng et al., 2025). CountGen is evaluated on SDXL using its official implementation, which relies on components specific to that backbone. Counting Guidance is evaluated using its original SD 1.4 implementation and our adaptations to FLUX.1-dev and SD 3.5 Large. YOLO-Count is evaluated on SDXL-Turbo (Sauer et al., 2024) following its published generation-control setting, which performs gradient-based token optimization with single-step generation.

Count Estimation and Evaluation. We use the pretrained open-vocabulary detector Grounding DINO (G-DINO) (Liu et al., 2024) both for intermediate count estimation in the feedback-based ATHENA strategies and as the primary evaluator of final count fidelity. The same detector configuration is used across all methods and datasets. Importantly, ATHENA is not tied to G-DINO: the intermediate count estimate determines the feedback control condition but does not directly define the steering direction. We further evaluate this dependence by replacing G-DINO with a vision-language model for intermediate count estimation (Section I.1) and independently evaluate final generations using Qwen3-VL-32B-Instruct (Bai et al., 2025)(Section 4.4).

Table 2: Counting performance across diffusion backbones and datasets. Accuracy (%) is reported as exact-match count accuracy, with the best result for each model–dataset pair shown in **bold**. MAE and RMSE measure counting error, and Time denotes mean generation time per sample (seconds).

Model	Method	CoCoCount				CoCoCount-E				ATHENA-COUNT			
		Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	Unsteered	58.4	0.98	1.96	45.8	46.5	1.12	1.93	22.6	39.4	1.78	2.96	45.5
	Counting Guidance	58.4	0.70	1.45	150.6	47.2	1.08	1.90	81.7	39.4	1.63	2.73	150.5
	ATHENA-Static	**73.3**	0.56	1.35	54.8	58.5	0.84	1.73	27.4	48.9	1.31	2.44	54.7
	ATHENA-Feedback	71.4	0.67	1.72	56.0	58.3	0.98	2.00	29.8	52.2	1.38	2.60	60.8
	ATHENA-Adaptive	70.2	0.65	1.63	64.8	**62.2**	0.85	1.90	35.1	**53.6**	1.40	2.72	72.5
SD 3.5 Large	Unsteered	58.4	0.78	1.50	43.9	44.8	0.98	1.56	24.1	38.9	1.55	2.54	43.8
	Counting Guidance	59.6	0.70	1.56	138.9	47.4	0.92	1.52	81.8	40.6	1.34	2.18	139.1
	ATHENA-Static	68.3	0.70	1.57	53.4	52.9	0.97	1.78	29.5	46.7	1.40	2.52	52.7
	ATHENA-Feedback	71.4	0.63	1.51	54.9	59.1	0.80	1.77	32.3	51.1	1.31	2.44	58.7
	ATHENA-Adaptive	**78.3**	0.41	1.05	62.0	**65.6**	0.73	1.60	38.1	**56.1**	1.16	2.32	70.4
SDXL	Unsteered	31.7	2.47	4.92	9.4	26.5	3.09	5.85	5.3	19.7	4.02	7.94	9.4
	CountGen	50.3	1.90	4.60	44.1	41.7	2.04	4.68	29.2	29.4	2.70	5.45	55.7
	ATHENA-Static	39.8	1.98	3.85	11.2	31.8	2.40	4.19	6.3	27.5	2.61	4.62	11.2
	ATHENA-Feedback	46.6	1.67	3.59	15.0	36.4	2.24	4.93	8.8	27.2	2.71	4.96	15.8
	ATHENA-Adaptive	**54.0**	1.50	3.47	19.4	**45.5**	1.93	4.52	11.6	**34.7**	2.34	3.90	21.2
SDXL Turbo	YOLO-Count	49.1	1.29	2.26	24.0	42.1	1.59	2.95	13.3	21.4	2.76	3.98	23.2
SD 1.4	Counting Guidance	28.6	2.19	3.62	19.9	19.4	2.80	4.27	11.1	10.8	3.69	4.85	19.6

Hyperparameter Selection. We tune the ATHENA hyperparameters, including the estimation step t_{est}, steering horizon t_{steer}, and steering strength γ, exclusively on CoCoCount and separately for each backbone. The selected values are then fixed and reused without modification for CoCoCount-E and the ATHENA-COUNT dataset. The hyperparameter values used in our experiments, together with an analysis of the sensitivity of ATHENA-Adaptive to the multiplicative scaling factor, are provided in Section F; performance remains robust across the tested scaling factors.

Computational Setup. We use NVIDIA RTX A6000 GPUs (48 GB) for CoCoCount and the ATHENA-COUNT dataset, and NVIDIA A100 SXM4 GPUs (40 GB) for CoCoCount-E. Within each dataset, all models and methods are evaluated using the same GPU type, enabling consistent runtime comparisons. Unless otherwise specified, all experiments use a fixed random seed. Because hardware differs across datasets, absolute generation times are not directly comparable across datasets; we therefore focus on relative accuracy–runtime trade-offs under fixed hardware.

4.2 MAIN COUNTING RESULTS

We first evaluate whether ATHENA improves count fidelity across the primary generative backbones and counting benchmarks. Table 2 summarizes the results. Across all model–dataset combinations, each ATHENA strategy improves exact-count accuracy over unsteered generation, demonstrating consistent gains from cardinality-specific test-time steering. ATHENA-Adaptive achieves the highest exact-count accuracy in eight of the nine settings, while ATHENA-Static performs best for FLUX.1-dev on CoCoCount. Seed sensitivity results are provided in Section G.1.

Compared with existing count-control methods, ATHENA substantially improves over Counting Guidance on both FLUX.1-dev and SD 3.5 Large across all three datasets. On SDXL, ATHENA-Adaptive also outperforms CountGen across all three datasets. Compared with YOLO-Count in its published SDXL-Turbo setting, ATHENA-Adaptive on SDXL achieves higher exact-count accuracy across all three datasets, despite the methods operating on different underlying generators.

We further analyze performance across target cardinalities in Section G.2, showing how count fidelity changes as the number of objects increases. Beyond count accuracy, we also examine whether these improvements affect the visual quality of the generated images. Across multiple perceptual metrics, ATHENA maintains comparable image quality to unsteered generation, while qualitative comparisons illustrate count corrections with largely preserved scene content and appearance; detailed image-quality results and qualitative examples are provided in Sections J.1 and J.2.

4.3 ACCURACY–EFFICIENCY TRADE-OFFS

We first examine the computational trade-offs among the ATHENA control strategies. We measure efficiency using end-to-end wall-clock generation time, from receiving the text prompt to producing the final image, including all intermediate computation required by each method; post-generation evaluation is excluded. As shown in Table 2, ATHENA-Static introduces only a small runtime

Table 3: Count accuracy (%) under G-DINO and Qwen3-VL-32B-Instruct evaluation. The best result for each evaluator and model–dataset pair is shown in **bold**.

Model	Method	CoCoCount		CoCoCount-E		ATHENA-COUNT	
		G-DINO	Qwen3-VL	G-DINO	Qwen3-VL	G-DINO	Qwen3-VL
FLUX.1-dev	Unsteered	58.4	60.87	46.5	50.00	39.4	47.22
	ATHENA-Static	**73.3**	**76.40**	58.5	62.81	48.9	56.67
	ATHENA-Feedback	71.4	71.43	58.3	62.96	52.2	56.67
	ATHENA-Adaptive	70.2	72.67	**62.2**	**66.82**	**53.6**	**59.17**
SD 3.5 Large	Unsteered	58.4	62.73	44.8	49.07	38.9	44.44
	ATHENA-Static	68.3	68.32	52.9	54.48	46.7	55.00
	ATHENA-Feedback	71.4	73.91	59.1	64.04	51.1	57.78
	ATHENA-Adaptive	**78.3**	**77.64**	**65.6**	**67.75**	**56.1**	**60.83**
SDXL	Unsteered	31.7	34.78	26.5	29.78	19.7	21.11
	ATHENA-Static	39.8	42.86	31.8	33.64	27.5	28.89
	ATHENA-Feedback	46.6	49.69	36.4	39.35	27.2	31.94
	ATHENA-Adaptive	**54.0**	**59.00**	**45.5**	**47.69**	**34.7**	**36.39**

overhead over unsteered generation, while Feedback and Adaptive require additional computation due to intermediate count estimation and additional diffusion steps. Across all evaluated models and datasets, however, the average number of additional steps remains below one full diffusion pass, with detailed statistics provided in Section H.

Compared with existing count-control methods, ATHENA provides favorable accuracy–runtime trade-offs across the evaluated settings. On FLUX.1-dev and SD 3.5 Large, ATHENA achieves higher counting accuracy than Counting Guidance while requiring substantially less generation time. On SDXL, ATHENA-Adaptive outperforms CountGen across all three datasets while achieving up to 2.5× faster generation. At a comparable runtime to Counting Guidance on SD 1.4, it achieves approximately 2× or higher counting accuracy, and it also achieves higher accuracy with lower runtime than YOLO-Count on SDXL-Turbo across all three datasets. These trade-offs are illustrated on the ATHENA-COUNT dataset in Figure 4, with results for the other datasets provided in Section H.

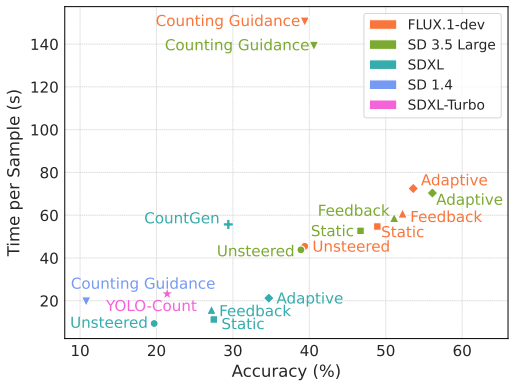


Figure 4: Accuracy–runtime trade-off on the ATHENA-COUNT. Colors denote the diffusion backbone, while marker shapes indicate the method.

Finally, ATHENA-Adaptive on SDXL requires 11.9 GB, compared with 17.2 GB for Counting Guidance on SD 1.4 and 47.5 GB for CountGen. This corresponds to approximately 4× lower memory requirement than CountGen, complementing the favorable accuracy–runtime trade-offs observed.

4.4 INDEPENDENT VALIDATION OF COUNT FIDELITY

Our primary evaluation uses G-DINO to provide a consistent automated measure of count fidelity across all methods, while ATHENA additionally uses G-DINO for intermediate count estimation during feedback-based steering (see Section I.1 for results using an alternative intermediate count estimator). To independently validate the reported improvements, we re-evaluate the same generated images from all primary model–dataset settings using Qwen3-VL-32B-Instruct. As shown in Table 3, Qwen3-VL generally reports higher absolute exact-count accuracy, while preserving the performance trends observed with G-DINO: all ATHENA strategies consistently improve over unsteered generation, and the best-performing strategy under G-DINO remains the best-performing strategy under Qwen3-VL across all settings. These results provide independent evidence that the observed gains in count fidelity are not specific to G-DINO-based final evaluation.

4.5 GENERALIZATION AND ROBUSTNESS

We first examine whether the counting challenge and the effectiveness of ATHENA extend to recent T2I models. As shown in Table 4, we apply ATHENA to FLUX.2-dev using the same hyperparameter

Table 4: Counting performance on recent T2I models across CoCoCount and the ATHENA-COUNT dataset. The best ATHENA result for each dataset is shown in **bold**.

Model	Method	CoCoCount			ATHENA-COUNT		
		Acc (↑)	MAE (↓)	RMSE (↓)	Acc (↑)	MAE (↓)	RMSE (↓)
FLUX.2-dev	Unsteered	63.98	0.602	1.249	38.10	1.430	2.320
	ATHENA-Static	70.19	0.497	1.069	49.17	1.172	2.177
	ATHENA-Feedback	**72.67**	0.478	1.238	47.22	1.294	2.371
	ATHENA-Adaptive	71.43	0.547	1.388	**52.22**	1.133	2.163
Qwen-Image	Unsteered	52.80	0.665	1.128	42.20	1.350	2.360

configuration selected for FLUX.1-dev, without additional tuning. Across both CoCoCount and the ATHENA-COUNT dataset, all three ATHENA strategies improve exact-count accuracy over unsteered FLUX.2-dev, showing that the proposed steering framework transfers to a newer generative backbone. We additionally evaluate unsteered Qwen-Image and observe substantial counting errors on both datasets, indicating that explicit count fidelity remains challenging in recent T2I models.

We next evaluate whether ATHENA extends to simultaneous count constraints for multiple object categories using CoCoCount-MO. We report *joint accuracy*, requiring all count constraints in a prompt to be satisfied, and *individual accuracy*, evaluating each constraint separately. We extend the feedback-based strategies by representing target, estimated, and error cardinalities as vectors over the specified objects; Adaptive determines directional consistency between successive corrections using the inner product of their error vectors.

Table 5: Multi-object counting performance on CoCoCount-MO. The best joint and individual accuracies for each evaluated backbone are shown in **bold**.

Model	Method	Joint Acc. (↑)	Individual Acc. (↑)	MAE (↓)	RMSE (↓)
FLUX.1-dev	Unsteered	11.0	30.2	1.815	2.702
	ATHENA-Static	**24.5**	**43.8**	1.505	2.482
	ATHENA-Feedback	17.0	38.8	1.750	2.911
	ATHENA-Adaptive	23.5	41.5	1.788	2.965
SD 3.5 Large	Unsteered	11.0	27.0	1.890	2.858
	ATHENA-Static	16.0	34.3	2.150	3.561
	ATHENA-Feedback	17.0	34.3	1.893	2.914
	ATHENA-Adaptive	**18.0**	**35.7**	1.835	2.856
SDXL	Unsteered	1.5	13.0	4.135	6.249

As shown in Table 5, ATHENA improves both accuracy measures over unsteered generation on SD 3.5 Large and FLUX.1-dev, with the largest joint-accuracy gain on FLUX.1-dev from 11.0% to 24.5%. Nevertheless, absolute joint accuracy remains relatively low, highlighting the difficulty of simultaneous cardinality control. Unsteered SDXL achieves only 1.5% joint accuracy; given this limited baseline capability, we do not apply the multi-object steering extension to this backbone.

Finally, we evaluate ATHENA on the counting component of GenEval (Ghosh et al., 2023) as an external standardized benchmark. ATHENA improves count accuracy over unsteered generation across all evaluated backbones, providing further evidence that its gains extend beyond our primary evaluation datasets; detailed results are provided in Section I.2.

5 CONCLUSION, LIMITATIONS, AND FUTURE DIRECTIONS

We introduced ATHENA, a model-agnostic, training-free framework for improving count fidelity in T2I diffusion models through cardinality-specific test-time steering. Across multiple generative backbones and counting benchmarks, including the challenging ATHENA-COUNT dataset introduced in this work, ATHENA consistently improves count fidelity while providing favorable accuracy–efficiency trade-offs and maintaining comparable visual quality. Experiments on recent T2I models and multi-object counting prompts further demonstrate the broader applicability of the framework. Nevertheless, ATHENA is designed specifically for cardinality control, and its feedback-based strategies depend on reliable intermediate count estimation, although the framework is not tied to a specific estimator. Extending this test-time steering approach beyond cardinality to directly target richer compositional constraints and semantic feedback signals represents a promising direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing baseline methods, generating candidate prompts for ATHENA-COUNT and CoCoCount-MO benchmarks, providing limited feedback on the theoretical formulation, and revising the manuscript. Generative AI was not used to originate the core research idea or construct the primary experimental pipeline. Also, all the final decisions regarding the methodology, analysis, and conclusions were made by the authors. The prompt-generation procedures are described in Section E, and all generated prompts were manually reviewed before inclusion. All AI-assisted code was reviewed and tested, while all AI-assisted manuscript content was manually checked and revised by the authors. We take full responsibility for the final content of this work, including all text, claims, code, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide the code and data used in this work at `https://anonymous.4open.science/r/ATHENA-8527`. The main paper describes the proposed method and experimental setup, while the appendix provides additional details needed to reproduce our results. In particular, detailed pseudocode for the ATHENA steering strategies is provided in Section D, the steering hyperparameters are reported in Section F, and the construction procedures, generation prompts, and statistics for ATHENA-COUNT and CoCoCount-E are documented in Section E. The corresponding code and data are included in the anonymous repository. Together, these materials provide the resources and implementation details necessary to reproduce the experiments and evaluations reported in this work.

REFERENCES

Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *The Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Lital Binyamin, Yoad Tewel, Hilit Segev, Eran Hirsch, Royi Rassin, and Gal Chechik. Make It Count: Text-to-Image Generation with an Accurate Number of Objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13242–13251, 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.

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.

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.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-Prompt Image Editing with Cross-Attention Control. In *The Eleventh International Conference on Learning Representations*, 2023.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Advances in neural information processing systems*, 30, 2017.

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.

Ivana Kajić, Olivia Wiles, Isabela Albuquerque, Matthias Bauer, Su Wang, Jordi Pont-Tuset, and Aida Nematzadeh. Evaluating numerical reasoning in text-to-image models. *Advances in Neural Information Processing Systems*, 37:42211–42224, 2024.

Wonjun Kang, Kevin Galim, Hyung Il Koo, and Nam Ik Cho. Counting guidance for high fidelity text-to-image synthesis. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 899–908. IEEE, 2025.

Andrey Kharitenko, Zebang Shen, Riccardo De Santi, Niao He, and Florian Dorfler. Landing with the score: Riemannian optimization through denoising. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=xZNoeX0z9f.

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.

Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024.

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

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. FLUX. 1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. *arXiv preprint arXiv:2506.15742*, 2025.

Joohyeon Lee, Jin-Seop Lee, and Jee-Hyong Lee. CountCluster: Training-Free Object Quantity Guidance with Cross-Attention Map Clustering for Text-to-Image Generation. *arXiv preprint arXiv:2508.10710*, 2025.

Yanyu Li, Pencheng Wan, Liang Han, Yaowei Wang, Liqiang Nie, and Min Zhang. CountDiffusion: Text-to-Image Synthesis with Training-Free Counting-Guidance Diffusion. *arXiv preprint arXiv:2505.04347*, 2025.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Anindya Mondal, Sauradip Nag, Ayan Banerjee, Josep Lladós, Xiatian Zhu, and Anjan Dutta. Countloop: Training-free high-instance image generation via iterative agent guidance. *Transactions on Machine Learning Research (TMLR)*, 2026.

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. *arXiv preprint arXiv:2307.01952*, 2023.

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.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Christoph Schuhmann. LAION Aesthetic Predictor. https://github.com/christophschuhmann/improved-aesthetic-predictor, 2022.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Jan Pawel Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Diffusion models encode the intrinsic dimension of data manifolds. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 46412–46440. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/stanczuk24a.html.

Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and Evolution of Interpretable Concepts in Diffusion Models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Binxu Wang and John J Vastola. Diffusion models generate images like painters: an analytical theory of outline first, details later. *arXiv preprint arXiv:2303.02490*, 2023.

Chen Wu and Fernando De la Torre. Contrastive prompts improve disentanglement in text-to-image diffusion models. *arXiv preprint arXiv:2402.13490*, 2024.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

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.

Oz Zafar, Yuval Cohen, Lior Wolf, and Idan Schwartz. Detection-Driven Object Count Optimization for Text-to-Image Diffusion Models. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1885–1894. IEEE, 2026.

Guanning Zeng, Xiang Zhang, Zirui Wang, Haiyang Xu, Zeyuan Chen, Bingnan Li, and Zhuowen Tu. YOLO-Count: Differentiable Object Counting for Text-to-Image Generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16765–16775, 2025.

A Diffusion Model Preliminaries

T2I diffusion models synthesize images by iteratively transforming noise into data through a sequence of denoising steps. At sampling step t, a learned denoiser $\epsilon_\theta(\cdot)$ is applied to the current latent representation z_t conditioned on a text prompt p. We denote the denoiser output by $\epsilon_t \triangleq \epsilon_\theta(t, z_t, p)$. The latent state is then updated using a sampler-specific transition operator,

$$z_{t-1} = \mathcal{S}_t(z_t, \epsilon_t, p), \tag{6}$$

where $\mathcal{S}_t(\cdot)$ may be stochastic or deterministic depending on the sampling scheme, encompassing diffusion samplers such as DDPM (Ho et al., 2020) and deterministic probability-flow variants (Song et al., 2021) used in modern systems.

The denoiser output also defines an estimate of the underlying clean latent at step t, denoted by $\hat{z}_0^{(t)}$, via a model-specific reconstruction operator

$$\hat{z}_0^{(t)} = \mathcal{D}_t(z_t, \epsilon_t). \tag{7}$$

The estimated clean latent may be decoded into image space for intermediate analysis when required.

B Detailed Geometric Motivation

From denoiser contrast to score contrast. Consider the Gaussian corruption process

$$z_t = \alpha_t z_0 + \sigma_t \xi, \qquad \xi \sim \mathcal{N}(0, I),$$

where z_0 is the clean latent, z_t is its noisy version at timestep t, and α_t and $\sigma_t > 0$ are determined by the noise schedule. For a conditioning prompt q, let $\epsilon_{t,q} \triangleq \epsilon_\theta(t, z_t, q)$ denote the denoiser prediction.

Under the standard mean-squared noise-prediction objective, the population-optimal denoiser predicts the conditional mean of the injected noise. We therefore write

$$\epsilon_{t,q} = \mathbb{E}[\xi \mid z_t, q] + r_{t,q},$$

where $r_{t,q}$ denotes the denoiser's approximation error.

Assuming that the conditional density $p_t(z_t \mid q)$ is differentiable, its score is

$$\begin{aligned} s_{t,q}(z_t) &\triangleq \nabla_{z_t} \log p_t(z_t \mid q) \\ &= \mathbb{E}[\nabla_{z_t} \log p(z_t \mid z_0) \,|\, z_t, q] \\ &= -\frac{1}{\sigma_t^2}\mathbb{E}[z_t - \alpha_t z_0 \mid z_t, q] = -\frac{1}{\sigma_t}\mathbb{E}[\xi \mid z_t, q]. \end{aligned}$$

Consequently,

$$-\frac{1}{\sigma_t}\left(\epsilon_{t,p} - \epsilon_{t,\hat{p}}\right) = s_{t,p}(z_t) - s_{t,\hat{p}}(z_t) - \frac{1}{\sigma_t}\left(r_{t,p} - r_{t,\hat{p}}\right).$$

If the difference between the prediction errors is small,

$$-\frac{1}{\sigma_t}\left(\epsilon_{t,p} - \epsilon_{t,\hat{p}}\right) \approx s_{t,p}(z_t) - s_{t,\hat{p}}(z_t). \tag{8}$$

This assumption concerns the difference between the prediction errors, rather than their individual magnitudes. Because both predictions use the same model, timestep, and latent state, and their prompts differ primarily in count, their errors are expected to be correlated and partially cancel, even if neither error is individually small.

From score contrast to manifold geometry. Let $\mathcal{M}_{t,q}$ denote the local high-probability-density manifold of latent states under $p_t(z_t \mid q)$. Importantly, $\mathcal{M}_{t,q}$ is defined at the same noise level t as z_t and is therefore distinct from the clean-image manifold. We assume that z_t lies in a neighborhood of $\mathcal{M}_{t,q}$ where its Euclidean projection

$$\Pi_{\mathcal{M}_{t,q}}(z_t) \triangleq \arg\min_{u \in \mathcal{M}_{t,q}} \|z_t - u\|_2$$

is locally unique.

Following the low-noise manifold approximation used in prior work (Alain & Bengio, 2014; Stanczuk et al., 2024; Kharitenko et al., 2026), we assume that the conditional density decreases approximately as an isotropic Gaussian with distance from $\mathcal{M}_{t,q}$. Locally,

$$p_t(z_t \mid q) \propto \rho_{t,q}\big(\Pi_{\mathcal{M}_{t,q}}(z_t)\big) \exp\left(-\frac{\left\|z_t - \Pi_{\mathcal{M}_{t,q}}(z_t)\right\|_2^2}{2\tau_{t,q}^2}\right), \tag{9}$$

where $\rho_{t,q}$ describes variation in probability density along the manifold and $\tau_{t,q}^2$ denotes the local variance normal to it.

Within a neighborhood having a unique projection,

$$\nabla_{z_t} \frac{1}{2}\left\|z_t - \Pi_{\mathcal{M}_{t,q}}(z_t)\right\|_2^2 = z_t - \Pi_{\mathcal{M}_{t,q}}(z_t).$$

If $\rho_{t,q}$ varies slowly relative to the normal Gaussian term, the tangential component of the score can be neglected. Taking the gradient of Equation (9) then gives

$$s_{t,q}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,q}}(z_t) - z_t}{\tau_{t,q}^2}. \tag{10}$$

Thus, the conditional score approximately points from z_t toward its local projection onto the condition-specific manifold.

Because both conditional distributions are evaluated at the same timestep, they share the noise-induced local variance τ_t^2. Subtracting their score approximations, therefore, yields

$$s_{t,p}(z_t) - s_{t,\hat{p}}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,p}}(z_t) - \Pi_{\mathcal{M}_{t,\hat{p}}}(z_t)}{\tau_t^2}. \tag{11}$$

If $\hat{p}$ is count-agnostic, $\mathcal{M}_{t,\hat{p}}$ contains modes associated with different object counts, including the count represented by the emerging image. Alternatively, if $\hat{p}$ specifies the actual count represented by z_t, it directly describes the corresponding count-specific region. In both cases, we assume that z_t is aligned with the remaining semantics of p. The control condition then locally describes the current latent state, giving

$$\Pi_{\mathcal{M}_{t,\hat{p}}}(z_t) \approx z_t. \tag{12}$$

Substituting Equation (12) into Equation (11) gives

$$s_{t,p}(z_t) - s_{t,\hat{p}}(z_t) \approx \frac{\Pi_{\mathcal{M}_{t,p}}(z_t) - z_t}{\tau_t^2}. \tag{13}$$

Hence, the score contrast approximately recovers the local residual from the current latent state toward the target-conditioned manifold.

Implication for ATHENA steering. Combining Equations (8) and (13) gives

$$\epsilon_{t,p} - \epsilon_{t,\hat{p}} \approx -\frac{\sigma_t}{\tau_t^2}\left(\Pi_{\mathcal{M}_{t,p}}(z_t) - z_t\right). \tag{14}$$

Thus, up to a negative scalar, the contrast between the denoiser predictions approximates the residual from the current latent state toward the target-conditioned manifold.

ATHENA modifies the target-conditioned prediction according to

$$\widetilde{\epsilon}_t = \epsilon_{t,p} + \gamma\left(\epsilon_{t,p} - \epsilon_{t,\hat{p}}\right), \tag{15}$$

where $\gamma \geq 0$ controls the steering strength. Under noise prediction, a denoiser output ϵ induces the clean-latent estimate

$$\widehat{z}_0(\epsilon) = \frac{z_t - \sigma_t \epsilon}{\alpha_t}. \tag{16}$$

Let $\widehat{z}_{0,p} = \widehat{z}_0(\epsilon_{t,p})$ and $\widetilde{z}_0 = \widehat{z}_0(\widetilde{\epsilon}_t)$. Substituting Equation (15) into Equation (16) gives

$$\widetilde{z}_0 - \widehat{z}_{0,p} = -\frac{\gamma\sigma_t}{\alpha_t}\left(\epsilon_{t,p} - \epsilon_{t,\hat{p}}\right). \tag{17}$$

Using Equation (14), this shift becomes

$$\widetilde{z}_0 - \widehat{z}_{0,p} \approx \frac{\gamma\sigma_t^2}{\alpha_t\tau_t^2}\left(\Pi_{\mathcal{M}_{t,p}}(z_t) - z_t\right). \tag{18}$$

Because $\gamma\sigma_t^2/(\alpha_t\tau_t^2)$ is nonnegative, the change in the predicted clean latent is aligned with the local residual toward the target-conditioned manifold. Therefore, under the stated denoiser-error, local-manifold, isotropy, and semantic-alignment assumptions, ATHENA's contrastive update moves the clean estimate in a direction that increases its consistency with the target count.

C Intermediate Count Estimation Analysis

At each sampling step, the model output induces an estimate of the final clean latent through the model-specific reconstruction operator $\mathcal{D}_t$. Given the current latent z_t and model prediction ϵ_t, the corresponding clean latent and image estimates are

$$\hat{z}_0^{(t)} = \mathcal{D}_t(z_t, \epsilon_t), \qquad \hat{x}_0^{(t)} = \mathrm{Dec}\left(\hat{z}_0^{(t)}\right). \tag{19}$$

Here, $\mathrm{Dec}(\cdot)$ denotes the latent decoder.

Although early reconstructions are incomplete, they improve progressively throughout sampling. Prior work suggests that generation follows a coarse-to-fine progression, with global composition and spatial layout emerging early and later steps primarily refining local appearance and texture (Hertz et al., 2023; Wang & Vastola, 2023; Tinaz et al., 2025). We therefore hypothesize that the object count in the final image can be predicted from an intermediate reconstruction.

To test this hypothesis, we generate images using prompts from CoCoCount (Binyamin et al., 2025) with 50 sampling steps and reconstruct the predicted clean image after step 20 for FLUX.1-dev and SD 3.5 Large and after step 30 for SDXL. We apply G-DINO (Liu et al., 2024) to both the intermediate reconstruction and the corresponding final image and compare the resulting counts using exact-match accuracy, MAE, and MSE. As reported in Table 1, the intermediate counts achieve high exact-match agreement and low numerical error relative to the final counts, supporting the use of intermediate reconstructions for estimating the count that emerges in the completed image.

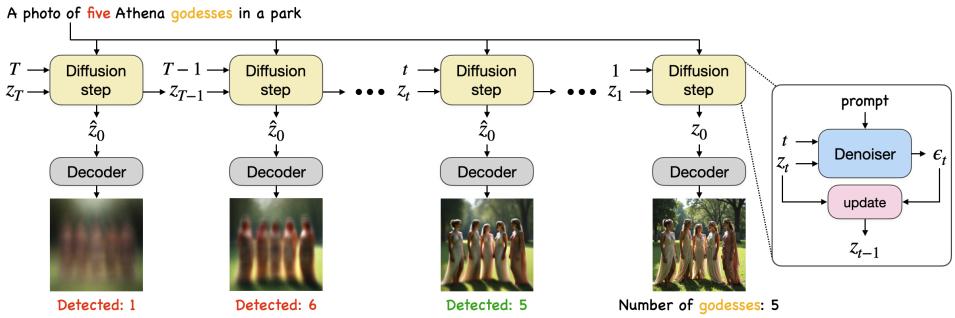


Figure 5: Count estimation across diffusion steps. Early decoded latents contain limited object structure for reliable counting, while later steps provide increasingly informative count estimates at higher inference cost. ATHENA performs count estimation at an intermediate diffusion step to balance reliability and efficiency.

Figure 5 further illustrates how the decoded clean-image estimate evolves throughout sampling, motivating count estimation at an intermediate sampling step to balance count observability and computational efficiency.

D ALGORITHMIC DETAILS OF ATHENA

D.1 ATHENA-STATIC

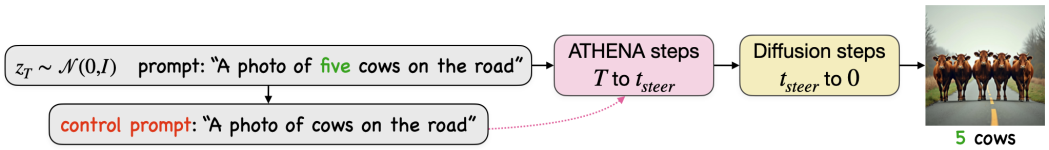


Figure 6: ATHENA-Static pipeline. Starting from the same initial noise, prompt-based steering is applied using a fixed, count-agnostic control prompt from diffusion step T down to a cutoff t_{steer}. Standard diffusion then proceeds unmodified to completion, yielding improved count fidelity without intermediate feedback or adaptive adjustment.

Algorithm 1 DIFFUSION-STEPS

Require: Denoiser: ϵ_θ, sampling operator: $\mathcal{S}_t$, text prompt: p, starting step: t_s, end step: t_e, initial latent: z_{t_s}

for $t = t_s$ to $t_e + 1$ **do**
  $\epsilon_t = \epsilon_\theta(t, z_t, p)$
  $z_{t-1} = \mathcal{S}_t(z_t, \epsilon_t)$
end for
Return: z_{t_e}

Algorithm 2 ATHENA-STEPS

Require: Denoiser: ϵ_θ, sampling operator: $\mathcal{S}_t$, text prompt: p, control prompt: $\hat{p}$, starting step: t_s, end step: t_e, initial latent: z_{t_s}, steering strength: $\gamma > 0$

for $t = t_s$ to $t_e + 1$ **do**
  $\epsilon_t = \epsilon_\theta(t, z_t, p)$
  $\hat{\epsilon}_t = \epsilon_\theta(t, z_t, \hat{p})$
  $\epsilon_{steer} = \epsilon_t + \gamma(\epsilon_t - \hat{\epsilon}_t)$
  $\epsilon_{steer} = \frac{\|\epsilon_t\|}{\|\epsilon_{steer}\|}\epsilon_{steer}$ {Matching the norm of the new update with the original update}
  $z_{t-1} = \mathcal{S}_t(z_t, \epsilon_{steer})$
end for
Return: z_{t_e}

Algorithm 3 ATHENA-STATIC

Require: Denoiser: ϵ_θ, sampling operator: $\mathcal{S}_t$, text prompt: p, target count: k, steering steps: t_{steer}, total steps: T, steering strength: $\gamma > 0$, decoder $\text{Dec}(\cdot)$

$z_T \sim \mathcal{N}(0, I)$
$\hat{p} \leftarrow$ remove k from p
$z_{t_{steer}} =$ **ATHENA-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, \hat{p}, t_s = T, t_e = t_{steer}, z_T, \gamma > 0)$
$z_0 =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = 0, z_{t_{steer}})$
$x = \text{Dec}(z_0)$
Return: x

D.2 ATHENA-FEEDBACK

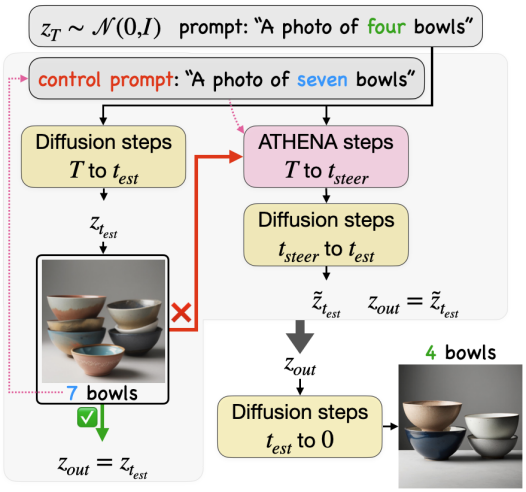


Figure 7: ATHENA-Feedback pipeline. An intermediate count is estimated at step t_{est} without steering. If the estimate differs from the target, generation is restarted from the same initial noise using a feedback control prompt, and a single early-stage steering phase is applied before completing diffusion.

Algorithm 4 ATHENA-FEEDBACK

Require: Denoiser: ϵ_θ, sampling operator: $\mathcal{S}_t$, reconstruction $\mathcal{D}_t$, text prompt: p, target count: k, steering steps: t_{steer}, estimation step t_{est}, total steps: T, steering strength: $\gamma > 0$, decoder $\text{Dec}(\cdot)$, counter $\text{Count}(\cdot)$

$z_T \sim \mathcal{N}(0, I)$
$\hat{p} \leftarrow$ remove k from p
$z_{t_{est}} =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = T, t_e = t_{est}, z_T)$
$\hat{z}_0 = \mathcal{D}_t(z_{t_{est}})$
$\hat{x} = \text{Dec}(\hat{z}_0)$
$c = \text{Count}(\hat{x})$
if $c = k$ **then**
  $z_{out} = z_{t_{est}}$
else
  $\hat{p} \leftarrow$ replace k in p with c
  $z_{t_{steer}} =$ **ATHENA-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, \hat{p}, t_s = T, t_e = t_{steer}, z_T, \gamma > 0)$
  $\tilde{z}_{t_{est}} =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{steer}})$
  $z_{out} = \tilde{z}_{t_{est}}$
end if
$z_0 =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{est}, t_e = 0, z_{out})$
$x = \text{Dec}(z_0)$
Return: x

D.3 ATHENA-ADAPTIVE

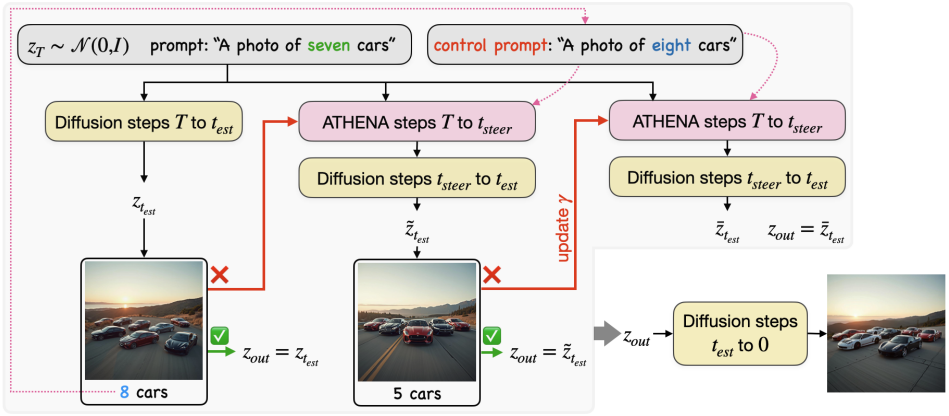


Figure 8: ATHENA-Adaptive pipeline. The method estimates object count at an intermediate diffusion step, applies early-stage prompt-based steering, adaptively adjusts the steering strength once based on the observed error direction, and completes generation using standard diffusion.

Algorithm 5 ATHENA-ADAPTIVE

Require: Denoiser: ϵ_θ, sampling operator: $\mathcal{S}_t$, reconstruction $\mathcal{D}_t$, text prompt: p, target count: k, steering steps: t_{steer}, estimation step t_{est}, total steps: T, steering strength: $\gamma > 0$, decoder $\text{Dec}(\cdot)$, counter $\text{Count}(\cdot)$

$z_T \sim \mathcal{N}(0, I)$
$\hat{p} \leftarrow$ remove k from p
$z_{t_{est}} =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = T, t_e = t_{est}, z_T)$
$\hat{z}_0 = \mathcal{D}_t(z_{t_{est}})$
$\hat{x} = \text{Dec}(\hat{z}_0)$
$c_1 = \text{Count}(\hat{x})$
if $c_1 = k$ **then**
  $z_{out} = z_{t_{est}}$
else
  $\hat{p} \leftarrow$ replace k in p with c_1
  $z_{t_{steer}} =$ **ATHENA-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, \hat{p}, t_s = T, t_e = t_{steer}, z_T, \gamma > 0)$
  $\tilde{z}_{t_{est}} =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{steer}})$
  $\hat{z}_0 = \mathcal{D}_t(\tilde{z}_{t_{est}})$
  $\hat{x} = \text{Dec}(\hat{z}_0)$
  $c_2 = \text{Count}(\hat{x})$
  if $c_2 = k$ **then**
    $z_{out} = \tilde{z}_{t_{est}}$
  else
    if $(c_1 \leq c_2 < k) \vee (k < c_2 \leq c_1)$ **then**
      {Case of not adding/removing enough objects}
      $\gamma \leftarrow 2 \times \gamma$
    else
      {Case of adding/removing too many objects}
      $\gamma \leftarrow \frac{\gamma}{2}$
    end if
    $z_{t_{steer}} =$ **ATHENA-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, \hat{p}, t_s = T, t_e = t_{steer}, z_T, \gamma > 0)$
    $\bar{z}_{t_{est}} =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{steer}})$
    $z_{out} = \bar{z}_{t_{est}}$
  end if
end if
$z_0 =$ **Diffusion-Steps** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{est}, t_e = 0, z_{out})$
$x = \text{Dec}(z_0)$
Return: x

E Dataset Construction and Statistics

E.1 CoCoCount and CoCoCount-E

The CoCoCount dataset Binyamin et al. (2025) consists of text prompts with explicit numerical constraints and minimal scene context, targeting object counts from 2 to 10. The original release includes prompts that differ only by random generation seed, as well as prompts with identical semantic structure but different target counts. As a result, the dataset contains repeated prompt content and heterogeneous count distributions.

For evaluation, we construct a deduplicated version of CoCoCount by retaining a single instance of each unique prompt text. This results in 161 distinct prompts, which we use as the CoCoCount benchmark in all experiments. While well-suited for controlled evaluation, CoCoCount offers limited diversity in prompt structure and scene complexity.

To enable controlled analysis across target cardinalities, we further construct *CoCoCount-E*, an extended benchmark derived from CoCoCount. Starting from a filtered subset of 72 unique prompt templates, we systematically instantiate each prompt with target counts from 2 to 10 while keeping all other prompt content fixed. This yields 648 prompts with consistent structure and a balanced count distribution.

Following the observation in prior work (Binyamin et al., 2025) that generation quality and counting accuracy degrade substantially beyond this range, we restrict the target count range to 2–10.

Table 6: Dataset statistics for CoCoCount and CoCoCount-E.

Dataset	Unique Prompts	Count Range	Total Prompts
CoCoCount (deduplicated)	161	2–10	161
CoCoCount-E	72	2–10	648

E.2 Athena-Count Dataset Generation

The Athena-Count dataset is generated by querying a large language model (GPT-5.2) using the prompt shown below. This prompt instructs the model to produce object-counting instructions with controlled variations in object category, scene context, distractors, and relational constraints, while sampling target counts between 2 and 10. Applying this prompt yields a structured collection of prompts organized into four difficulty levels, with consistent formatting and systematic increases in complexity. Finally, all generated samples are manually inspected to ensure quality.

Task Definition

```
Generate a challenging object-counting evaluation dataset for
↪  text-to-image diffusion models. The dataset is intended to
↪  evaluate count fidelity only, under increasingly difficult prompt
↪  conditions.

You are given:
- A JSON file containing the original CoCoCount dataset (attached).
- The specifications below.

You must generate a new dataset that:
- Does NOT reuse any object categories appearing in CoCoCount for
↪  Level 1.
- Uses clear, unambiguous noun-phrase prompts suitable for automatic
↪  counting.
- Contains no duplicate prompts.
- Contains exactly one counted object category per prompt.
- All prompts must begin with: "A photo of ..."
```

Dataset Output Format

```
- Output a single JSON file as a list of entries.
- Each entry must contain the following required fields.

Required JSON template (exact keys required):
```

```
{
  "id": <unique integer>,
  "prompt": <string>,                  // full prompt, starts with
  ↪  "A photo of"
  "object": <string>,                  // singular object name
  "object_plural": <string>,           // plural form used in the
  ↪  prompt
  "number": <string>,                  // number in words (e.g.,
  ↪  "three")
  "int_number": <integer>,                  // numeric count (e.g.,
  ↪  3)

  "level": <integer>,                  // 1, 2, 3, or 4
  "difficulty_tag": <string>,          // short descriptor (e.g.,
  ↪  "hard-object", "scene", "distractor", "relation")

  "scene": <string | null>,            // scene phrase if present
  ↪  (e.g., "on the ground")
  "distractor": <string | null>,       // uncounted object if
  ↪  present
  "relation": <string | null>,         // relational phrase if
  ↪  present

  "source": "ATHENA"
}
```

Detector Compatibility Constraint

```
All object categories MUST be reliably detectable by open-vocabulary
↪  grounding models
(e.g., Grounding DINO) under standard confidence thresholds.

Objects must:
- be visually salient at typical image resolutions,
- have strong and common visual-text associations,
- be distinguishable without fine-grained or microscopic detail,
commonly appear in natural images rather than technical diagrams or
↪  product-only photos.

Avoid objects that are:
- extremely small or thin,
- rare, niche, or domain-specific,
- visually indistinguishable without context,
- typically embedded inside other objects.
```

Count Constraints

```
- Valid counts: 2 - 10
- Counts should be approximately balanced
- "number" must be the word form of "count"
- "object_plural" must match the prompt text exactly
```

Difficulty Levels

Level 1: Hard Object Categories

```
- Object categories must NOT appear in CoCoCount.
- Objects should be visually challenging for counting (e.g., instance
↪  ambiguity, moderate occlusion, reflective or deformable surfaces),
↪  but must remain clearly recognizable as distinct object instances
↪  by open-vocabulary detection models.
- No verbs
- No scene
- No distractors
```

Prompt format:

```
A photo of <number> <object_plural>
```

Example pattern: *A photo of seven kites*

Level 2: Hard Objects + Scene Context

```
- Same object constraints as Level 1.
- Add a scene phrase.
- Scene must not introduce additional countable objects.
- No verbs
- No distractors
```

Prompt format:

```
A photo of <number> <object_plural> <scene>
```

Example pattern: *A photo of six lanterns in a temple hall*

Level 3: Counted Object + Semantic Distractor

```
- One counted object category.
- One uncounted distractor (no number specified).
- No verbs
- Avoid ambiguity about which object is counted.
```

Prompt format:

```
A photo of <number> <object_plural> <scene> with <distractor>
```

Example pattern: *A photo of five apples on a table with a dog*

Level 4: Relational Language

```
- One counted object category.
- Introduce spatial or relational structure.
- No second counted object.
- Relational phrasing may include light verb-based constructions if
↪  unavoidable.
```

Prompt format:

```
A photo of <number> <object_plural> <relation>
```

Example pattern: *A photo of eight chairs arranged around a round table*

Strict Constraints

```
- Do NOT reuse Level-1 object categories from CoCoCount.
- Do NOT include multiple counted objects.
- Do NOT use vague quantifiers ("several", "many").
```

```
- Do NOT use negation-based counting.
- Do NOT generate prompts requiring subjective interpretation.
- Do NOT generate near-duplicate prompts.
- Ensure "scene", "distractor", and "relation" are null when not
↪  applicable.
```

Dataset Size

```
Generate exactly 360 prompts, distributed as:
- Level 1: 120
- Level 2: 100
- Level 3: 100
- Level 4: 40
```

Output Requirements

```
- Output valid JSON only
- All required fields must be present
- Prompts must be natural, fluent, and concise
- Share the dataset link when you finish generating it
```

E.3 ATHENA-COUNT DATASET DETAILS

The ATHENA-COUNT consists of 360 prompts organized into four levels of increasing difficulty. Each level incrementally introduces additional complexity to the counting task. Prompts are generated using a large language model to systematically combine multiple constraints within a single instruction, enabling controlled diversity across difficulty levels. All generated prompts are manually reviewed to remove duplicates, improve linguistic clarity, and ensure compatibility with open-vocabulary object detectors. The four difficulty levels and their corresponding prompt structures are summarized in Table 7.

Table 7: Structure of the ATHENA-COUNT with four levels of increasing difficulty.

Level	Description	Example Prompt
L1	Hard object categories	"nine microphones"
L2	+ Scene context	"eight sneakers at the edge of a pond"
L3	+ Distractor objects	"five jars beside a river with a person"
L4	+ Relational constraints	"seven dumbbells lined up along a sidewalk"

We define *hard object categories* as object classes that are visually challenging for reliable instance counting due to factors such as high intra-class variability, deformable structure, partial occlusion, reflective surfaces, or weak canonical appearance. Unlike common object categories (e.g., "apples", "dogs"), these objects typically benefit less from strong visual priors learned during large-scale diffusion training, resulting in a more challenging benchmark for count-controlled generation.

Level 1: Hard Object Categories (120 prompts). This level focuses on visually challenging object categories with high variability or weak canonical structure. Example prompts include: *"A photo of nine dumbbells," "A photo of ten accordions," "A photo of eight telescopes," "A photo of seven hourglasses."*

Level 2: Scene Context (100 prompts). This level introduces environmental context, requiring models to localize objects within diverse scenes. Example prompts include: *"A photo of four saxophones at the edge of a pond," "A photo of three flashlights near a campfire," "A photo of five accordions in a museum gallery," "A photo of nine paddles on a marble countertop."*

Level 3: Distractor Objects (100 prompts). This level introduces additional non-target objects that may interfere with counting. Example prompts include: *"A photo of six jars beside a river with*

a person," "A photo of five lanterns on a street with cars," "A photo of seven helmets near bicycles," "A photo of eight globes next to books."

Level 4: Relational Constraints (40 prompts). This level introduces spatial or relational structure between objects, increasing compositional complexity. Example prompts include: *"A photo of seven dumbbells lined up along a sidewalk," "A photo of six lanterns arranged in a circle," "A photo of five helmets stacked on top of each other," "A photo of eight candles placed in two rows."*

F HYPERPARAMETER SETTINGS

F.1 HYPERPARAMETER VALUES

Unless otherwise specified, all single-seed experiments in the main paper use a fixed random seed (23) for image generation to ensure reproducibility. Table 8 summarizes the hyperparameter settings used for ATHENA across diffusion backbones and steering variants.

Table 8: Hyperparameter settings used for ATHENA across diffusion backbones and steering variants.

Parameter	**Diffusion Backbone**								
	FLUX.1-dev			**SD 3.5 Large**			**SDXL**		
	Static	Feedback	Adaptive	Static	Feedback	Adaptive	Static	Feedback	Adaptive
t_{est}	–	20	20	–	20	20	–	30	30
t_{steer}	10	5	5	10	5	5	10	10	10
γ	4	4	4	4	4	4	5	5	5

F.2 ADAPTIVE UPDATE SENSITIVITY

We evaluate the sensitivity of the adaptive update rule in ATHENA-Adaptive to the choice of multiplicative scaling factor used in updating γ. In addition to the default factor of 2 used in all main experiments, we consider alternative scaling factors of 1.5 and 4. Results on the ATHENA-COUNT dataset are reported in Table 9.

Table 9: Effect of the scaling factor in the adaptive update rule on the ATHENA-COUNT dataset.

Model	**Scaling Factor**	**Acc (↑)**	**MAE (↓)**	**RMSE (↓)**	**Time (↓)**
FLUX.1-dev	1.5	54.4	1.48	2.76	72.2
	2	53.6	1.40	2.72	72.5
	4	52.8	1.44	2.75	72.8
SD 3.5 Large	1.5	52.8	1.24	2.36	70.4
	2	56.1	1.16	2.32	70.4
	4	54.2	1.27	2.45	70.8
SDXL	1.5	36.9	2.31	3.99	21.5
	2	34.7	2.34	3.90	21.2
	4	34.2	2.57	4.65	21.4

We observe that performance remains consistent across the tested scaling factors for all three backbones, indicating that the adaptive mechanism is not highly sensitive to the exact choice of scaling factor.

G Additional Analysis of Counting Performance

G.1 Seed Sensitivity Analysis

To assess the robustness of our results to random initialization, we evaluate all methods across three seeds $\{7, 23, 42\}$ and report mean $\pm$ standard deviation in Table 10. All sources of randomness arise from the stochastic diffusion sampling process (i.e., the initial noise seed), while the test-time steering procedure itself is deterministic given a sampled trajectory as the models in evaluations used deterministic sampling.

Overall, ATHENA demonstrates stable performance across seeds, with relatively low variance in both accuracy and error metrics across all datasets and backbones. Importantly, the improvements over unsteered generation and prior baselines remain consistent under different seeds. The observed standard deviations are small relative to the performance gains, indicating that the improvements are not driven by favorable random initialization.

These results confirm that the performance benefits of ATHENA are robust and reproducible across different random seeds.

Table 10: Quantitative counting performance (mean $\pm$ std over seeds $\{7, 23, 42\}$). Accuracy (%) is reported as exact-match count accuracy. MAE and RMSE measure counting error, and Time denotes mean generation time per sample (seconds).

Model	Method	CoCoCount				CoCoCount-E				ATHENA-COUNT			
		Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	Unsteered	58.0±0.7	0.92±0.08	1.80±0.22	45.4±0.4	47.8±1.3	1.09±0.05	1.88±0.10	22.8±0.2	38.5±0.8	1.81±0.05	3.00±0.04	45.4±0.2
	ATHENA-Static	69.8±2.8	0.59±0.05	1.42±0.06	54.6±0.1	58.3±0.2	0.81±0.03	1.70±0.02	27.4±0.2	49.1±0.9	1.27±0.06	2.33±0.12	54.9±0.1
	ATHENA-Feedback	66.4±3.6	0.76±0.10	1.80±0.18	55.8±0.6	59.0±1.0	0.96±0.02	2.06±0.13	29.6±0.1	50.0±1.9	1.43±0.05	2.62±0.04	60.6±0.1
	ATHENA-Adaptive	66.7±3.2	0.75±0.13	1.79±0.23	64.0±1.8	62.4±0.2	0.85±0.00	1.90±0.03	34.9±0.1	54.5±0.9	1.34±0.08	2.57±0.16	72.9±0.5
SD 3.5 Large	Unsteered	58.4±1.0	0.78±0.06	1.54±0.18	44.3±0.3	47.0±1.7	0.94±0.04	1.57±0.08	24.2±0.0	38.1±0.7	1.51±0.05	2.47±0.07	44.0±0.2
	ATHENA-Static	65.4±2.0	0.72±0.06	1.64±0.15	53.3±0.4	53.2±0.8	0.96±0.02	1.79±0.12	29.2±0.2	46.4±1.5	1.26±0.09	2.40±0.12	52.8±0.2
	ATHENA-Feedback	68.9±1.7	0.62±0.02	1.42±0.10	54.7±0.6	61.3±2.7	0.76±0.05	1.55±0.12	32.1±0.5	51.4±1.0	1.28±0.04	2.39±0.02	58.9±0.1
	ATHENA-Adaptive	74.9±2.6	0.50±0.09	1.26±0.30	61.8±0.7	67.4±1.5	0.81±0.03	1.70±0.11	36.6±0.5	56.8±0.8	1.12±0.05	2.21±0.09	70.5±0.3
SDXL	Unsteered	31.3±0.3	2.32±0.25	4.75±1.30	9.3±0.0	24.7±1.7	2.89±0.23	5.44±0.49	5.3±0.0	24.9±1.8	3.29±0.36	6.20±1.15	5.3±0.0
	CountGen	48.2±2.9	2.02±0.13	4.65±0.06	45.5±1.3	40.9±0.4	2.11±0.08	4.58±0.08	45.8±0.7	29.6±0.7	2.79±0.11	5.56±0.17	53.4±1.3
	ATHENA-Static	40.2±1.3	2.16±0.17	4.59±0.87	11.3±0.0	29.9±1.5	2.71±0.15	4.99±0.25	6.3±0.0	29.2±1.2	2.59±0.13	4.66±0.47	11.2±0.1
	ATHENA-Feedback	47.0±2.5	2.00±0.31	4.65±1.11	14.9±0.2	36.3±0.6	2.48±0.24	5.64±0.63	8.8±0.0	27.6±1.1	2.70±0.05	4.96±0.01	15.1±0.1
	ATHENA-Adaptive	51.7±3.9	1.83±0.24	4.68±1.26	19.4±0.1	42.9±1.7	2.10±0.14	4.91±0.47	11.8±0.2	33.6±2.8	2.78±0.35	5.56±1.17	21.4±0.1
SD 1.4	Counting Guidance	25.3±2.9	2.49±0.41	4.03±0.91	20.1±0.5	16.4±2.9	3.18±0.38	4.70±0.80	14.4±0.4	10.5±2.1	3.78±0.09	4.88±0.08	19.7±0.1

G.2 Robustness Across Target Counts

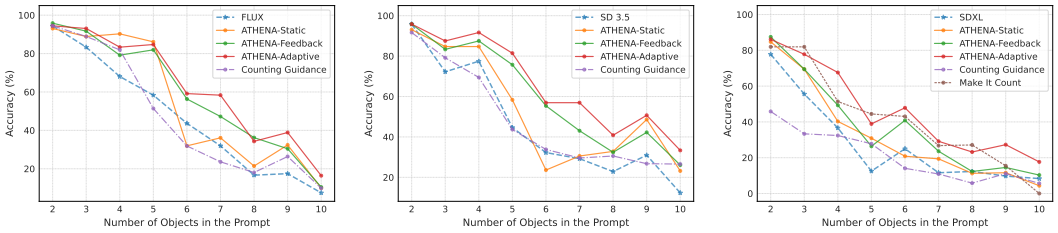
We analyze counting accuracy as a function of the target object count. Figure 9 reports results on CoCoCount-E and ATHENA-COUNT across the three primary diffusion backbones. Across these settings, counting accuracy generally decreases as the target count increases, reflecting the growing difficulty of enforcing precise cardinality.

As an example, Figure 9(c) shows the results on CoCoCount-E with the SDXL backbone. Unsteered generation degrades sharply beyond three objects, and prior baselines exhibit limited robustness as counts increase. In contrast, ATHENA-Adaptive maintains higher accuracy across the full count range. It achieves close to 80% accuracy for two to three objects and nearly $2\times$ the accuracy of unsteered sampling for larger counts.

H Additional Accuracy–Efficiency Analysis

Additional accuracy–runtime comparisons across models and datasets are shown in Figure 10.

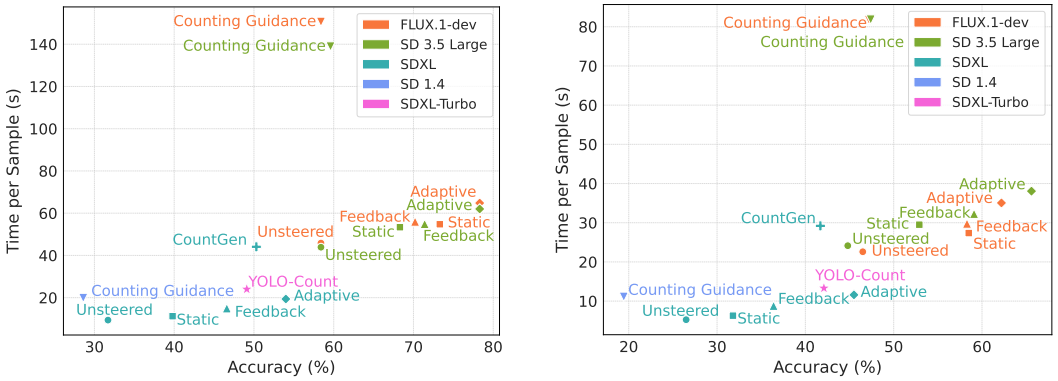
To provide additional insight into the computational overhead introduced by test-time steering, we report the average number of additional diffusion steps incurred by each method. While ATHENA-Static does not introduce extra steps, ATHENA-Feedback and ATHENA-Adaptive perform intermediate count estimation and may trigger additional steps depending on the feedback signal. This analysis quantifies the trade-off between improved count fidelity and computational overhead by measuring the number of additional diffusion steps, which reflects the algorithmic cost of the method and remains independent of hardware-specific runtime variations.



(a) CoCoCount-E (FLUX) (b) CoCoCount-E (SD 3.5) (c) CoCoCount-E (SDXL)

(d) ATHENA-COUNT (FLUX) (e) ATHENA-COUNT (SD 3.5) (f) ATHENA-COUNT (SDXL)

Figure 9: Counting accuracy versus target object count across datasets and diffusion backbones. ATHENA-Adaptive consistently maintains higher accuracy as the target count increases, demonstrating improved robustness to increasing counting difficulty.



(a) CoCoCount (b) CoCoCount-E

Figure 10: Accuracy–runtime trade-offs on CoCoCount and CoCoCount-E. ATHENA-Adaptive consistently achieves higher accuracy at comparable or lower runtime than prior baselines.

Table 11 summarizes the average number of additional steps across models and datasets. We observe that weaker backbones tend to incur more additional steps, likely reflecting the increased difficulty of achieving accurate counting with lower-capacity models. Overall, ATHENA introduces a moderate number of additional steps, consistent with the runtime overhead reported in the main paper. The additional steps remain below a full diffusion pass on average, and no extra steps are incurred when the initial generation already satisfies the target count.

I ROBUSTNESS AND GENERALIZATION

I.1 ALTERNATIVE INTERMEDIATE COUNT ESTIMATION

To evaluate the dependence of our method on the choice of count estimator, we replace G-DINO with a large vision language model (GPT) for intermediate count estimation in the Feedback and Adaptive variants. In this setting, GPT is prompted to estimate the number of target objects from generated images, and its output is used to curate the control prompt.

Table 11: Average number of additional diffusion steps introduced by each method.

Model	Method	CoCoCount	CoCoCount-E	ATHENA-COUNT
FLUX.1-dev	Static	0.0	0.0	0.0
	Feedback	8.7	10.8	12.6
	Adaptive	14.8	18.2	22.1
SD 3.5 Large	Static	0.0	0.0	0.0
	Feedback	8.6	11.9	12.5
	Adaptive	13.3	19.5	21.4
SDXL	Static	0.0	0.0	0.0
	Feedback	20.9	22.2	24.3
	Adaptive	36.3	39.3	44.8

In this experiment, only the intermediate count estimator is replaced, while final evaluation metrics are consistently computed using G-DINO across all methods. This isolates the effect of the estimator without altering the evaluation protocol. Results are summarized in Table 12.

Table 12: Effect of replacing the intermediate count estimator with GPT for the Feedback and Adaptive variants of ATHENA on CoCoCount and the ATHENA-COUNT dataset.

Model	Method	Estimator	CoCoCount				ATHENA-COUNT			
			Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	ATHENA-Feedback	G-DINO	71.4	0.67	1.72	56.0	52.2	1.38	2.60	60.8
	ATHENA-Feedback	GPT	68.3	0.78	1.84	58.5	48.1	1.54	2.76	62.8
	ATHENA-Adaptive	G-DINO	70.2	0.65	1.63	64.8	53.6	1.40	2.72	72.5
	ATHENA-Adaptive	GPT	68.3	0.69	1.54	66.5	53.1	1.47	2.79	75.4
SD 3.5 Large	ATHENA-Feedback	G-DINO	71.4	0.63	1.51	54.9	51.1	1.31	2.44	58.7
	ATHENA-Feedback	GPT	68.9	0.62	1.32	58.7	52.5	1.31	2.48	61.4
	ATHENA-Adaptive	G-DINO	78.3	0.41	1.05	62.0	56.1	1.16	2.32	70.4
	ATHENA-Adaptive	GPT	73.9	0.57	1.45	67.4	56.9	1.19	2.39	74.0

We observe that both Feedback and Adaptive variants maintain comparable performance when using GPT as the intermediate estimator across all evaluated backbones. These results indicate that the effectiveness of the method is not specific to G-DINO as the intermediate counting module. While GPT provides a strong closed-source alternative, GroundingDINO offers a fast and efficient open-source solution with negligible runtime overhead relative to the diffusion process.

I.2 EVALUATION ON GENEVAL

We evaluate ATHENA on the counting component of GenEval (Ghosh et al., 2023), comprising 80 prompts, to assess robustness under a standardized external benchmark. GenEval provides a standardized detection-based evaluation protocol for compositional image generation, and we follow its default evaluation protocol and detector configuration. In addition to the standard GenEval accuracy metric, we report mean absolute error (MAE) and root mean squared error (RMSE) computed from the predicted object counts, along with generation time per sample. Detailed results are reported in Table 13.

As shown in Table 13, ATHENA improves accuracy over unsteered generation across all evaluated backbones. On FLUX.1-dev, the best ATHENA result improves accuracy from 67.5% to 86.3%, while on SDXL it increases from 48.8% to 68.8%. ATHENA also achieves higher accuracy than the evaluated prior methods, CountGen and Counting Guidance, in their respective backbone settings. Overall, these results demonstrate that the improvements in count fidelity extend to the standardized GenEval counting benchmark beyond the datasets used in our primary evaluation.

Table 13: Results on the GenEval counting component. Accuracy (%) is reported using the GenEval protocol, with the best result for each model shown in **bold**. MAE and RMSE are computed from the predicted counts produced by the GenEval evaluation pipeline. Time denotes mean generation time per sample (seconds).

Model	Method	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	Unsteered	67.5	0.49	1.01	45.0
	ATHENA-Static	**86.3**	0.29	0.83	54.5
	ATHENA-Feedback	78.8	0.41	0.99	53.6
	ATHENA-Adaptive	81.3	0.40	1.00	57.8
SD 3.5 Large	Unsteered	73.8	0.28	0.65	43.9
	ATHENA-Static	**85.0**	0.20	0.67	53.2
	ATHENA-Feedback	80.0	0.28	0.69	51.2
	ATHENA-Adaptive	81.3	0.28	0.69	53.8
SDXL	Unsteered	48.8	0.55	1.13	9.3
	CountGen	57.5	0.79	1.47	48.6
	ATHENA-Static	53.8	0.60	1.20	11.1
	ATHENA-Feedback	67.5	0.49	1.14	13.2
	ATHENA-Adaptive	**68.8**	0.43	0.92	15.2
SD 1.4	Counting Guidance	18.8	1.65	2.15	19.6

J IMAGE QUALITY AND QUALITATIVE RESULTS

J.1 IMAGE QUALITY EVALUATION

We evaluate the impact of test-time steering on visual quality using a set of complementary perceptual metrics. While the main paper focuses on count fidelity, it is important to verify that improvements in counting accuracy do not degrade the visual quality of generated images.

We consider four widely-used reference-free metrics. CLIPScore Hessel et al. (2021) measures text-image alignment using CLIP embeddings. PickScore Kirstain et al. (2023) and HPSv2 Wu et al. (2023) are learned preference models trained to approximate human judgments of image quality. The LAION aesthetic score Schuhmann (2022) evaluates visual appeal using a model trained on human aesthetic ratings. Together, these metrics capture both semantic alignment and perceptual quality. All metrics are computed using publicly available pretrained models with default settings and are applied consistently across all methods and datasets.

We do not report FID Heusel et al. (2017), as it requires reference ground-truth images for each prompt, which are not available in our evaluation setting. Instead, we rely on reference-free metrics that are better suited for text-to-image generation.

Results are reported in Table 14. We observe that ATHENA maintains comparable visual quality to the unsteered baseline across all evaluated backbones and datasets, indicating that improvements in count fidelity do not come at the cost of visual quality.

J.2 QUALITATIVE RESULTS

Figures 11–14 present comparisons between unsteered generation, baselines, and ATHENA across diverse prompts. As shown, ATHENA improves count fidelity while preserving scene structure, object appearance, background coherence, and color consistency. Unlike prior baselines, ATHENA avoids visual artifacts such as background blurring, object blending, or disrupted spatial layout, and maintains coherent object–scene relationships in the shown examples, including cases with complex distractors and relational constraints.

In contrast, prior methods often modify scene content or introduce visual distortions when attempting to enforce counts. CountGen alters scene semantics, while Counting Guidance often fails to correct miscounting and degrades visual quality for complex prompts. Among the variants, ATHENA-Adaptive most consistently resolves counting errors, particularly for higher target counts and dense

Table 14: Image quality evaluation across diffusion backbones and datasets. All metrics are higher-is-better. CLIPScore measures text-image alignment, PickScore and HPSv2 approximate human preference, and Aesthetic evaluates visual appeal.

Model	Method	CoCoCount				CoCoCount-E				ATHENA-COUNT			
		CLIP (↑)	Aesthetic (↑)	PickScore (↑)	HPS (↑)	CLIP (↑)	Aesthetic (↑)	PickScore (↑)	HPS (↑)	CLIP (↑)	Aesthetic (↑)	PickScore (↑)	HPS (↑)
FLUX.1-dev	Unsteered	0.3601	5.71	23.94	0.3220	0.3662	5.74	24.01	0.3198	0.3807	5.61	24.18	0.3164
	Counting Guidance	0.3660	5.64	23.96	0.3253	0.3678	5.66	24.02	0.3233	0.3834	5.55	24.15	0.3135
	ATHENA-Static	0.3620	5.65	23.87	0.3258	0.3669	5.69	23.91	0.3245	0.3862	5.54	24.06	0.3159
	ATHENA-Feedback	0.3589	5.73	23.87	0.3213	0.3655	5.73	23.94	0.3208	0.3815	5.61	24.14	0.3172
	ATHENA-Adaptive	0.3607	5.73	23.86	0.3217	0.3659	5.73	23.95	0.3209	0.3816	5.59	24.10	0.3168
SD 3.5 Large	Unsteered	0.3721	5.44	23.33	0.3114	0.3789	5.42	23.33	0.3110	0.3983	5.33	23.74	0.3099
	Counting Guidance	0.3773	5.47	23.26	0.3120	0.3805	4.47	23.27	0.3114	0.3987	5.34	23.65	0.3085
	ATHENA-Static	0.3674	5.37	22.69	0.2921	0.3708	5.38	22.66	0.2929	0.3890	5.20	22.97	0.2837
	ATHENA-Feedback	0.3735	5.41	23.29	0.3101	0.3812	5.42	23.27	0.3094	0.3965	5.32	23.53	0.3042
	ATHENA-Adaptive	0.3740	5.41	23.24	0.3090	0.3792	5.42	23.21	0.3077	0.3959	5.16	23.39	0.2998
SDXL	Unsteered	0.3635	5.43	23.25	0.2978	0.3690	5.41	23.31	0.2959	0.3803	5.32	23.31	0.2848
	CountGen	0.3739	5.60	23.15	0.2921	0.3767	5.59	23.21	0.2909	0.3775	5.40	23.09	0.2764
	ATHENA-Static	0.3651	5.53	23.08	0.2967	0.3651	5.49	23.08	0.2963	0.3668	5.33	23.05	0.2809
	ATHENA-Feedback	0.3641	5.43	23.26	0.2967	0.3696	5.42	23.30	0.2962	0.3720	5.37	23.15	0.2837
	ATHENA-Adaptive	0.3692	5.41	23.34	0.2996	0.3697	5.43	23.29	0.2968	0.3715	5.36	23.09	0.2834
SD 1.4	Counting Guidance	0.3081	4.79	21.07	0.2393	0.3198	4.87	21.27	0.2427	0.3024	4.81	20.95	0.2219

layouts. Failure cases are rare and typically arise from inaccurate intermediate detection in especially challenging scenes. Overall, these results indicate that ATHENA enables accurate object counting at test time without sacrificing visual fidelity.

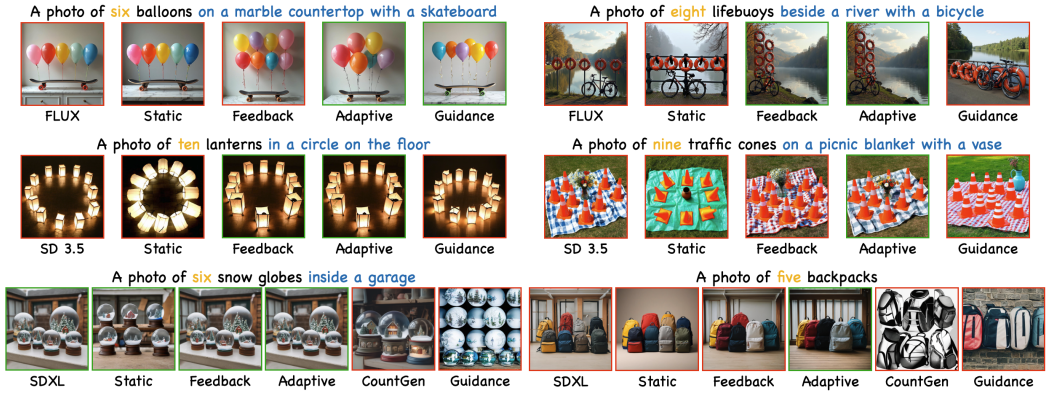


Figure 11: Qualitative results on complex, distractor-rich prompts. *Guidance* denotes Counting Guidance. Green borders indicate correct counts; red borders indicate incorrect counts. All ATHENA variants preserve structure and consistency while handling relational and multi-object instructions.

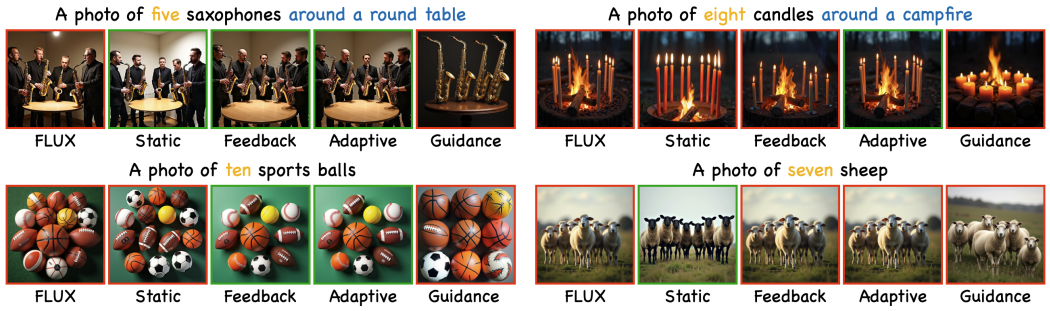


Figure 12: Additional qualitative results on relational and multi-object prompts. Results are shown for unsteered generation (FLUX) and the three ATHENA variants. Green borders indicate correct object counts, while red borders denote counting errors. ATHENA improves count fidelity across diverse settings (e.g., grouping, circular layouts, and natural scenes) while preserving scene structure and visual coherence, with the adaptive variant yielding the most consistent corrections.

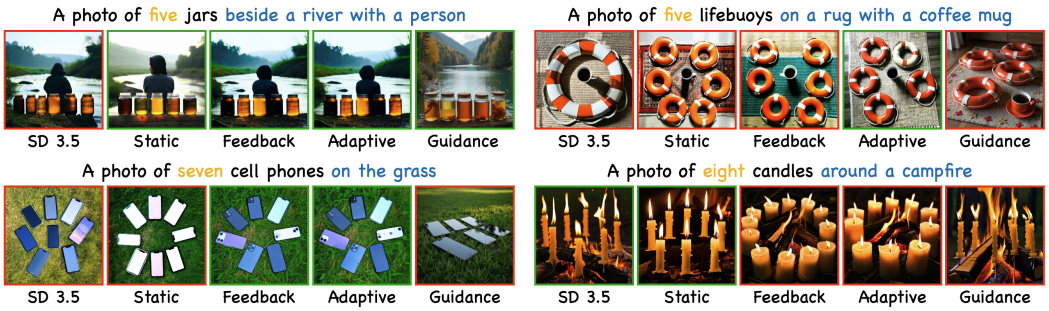


Figure 13: Additional qualitative results on relational and multi-object prompts. Results are shown for unsteered generation (SD 3.5) and the three ATHENA variants. Green borders indicate correct object counts, while red borders denote counting errors. ATHENA improves count fidelity across diverse settings (e.g., grouping, circular layouts, and natural scenes) while preserving scene structure and visual coherence, with the adaptive variant yielding the most consistent corrections.



Figure 14: Additional qualitative results on relational and multi-object prompts with the SDXL backbone. Results include unsteered generation, CountGen, Counting Guidance (*Guidance*), and the three ATHENA variants. Green borders indicate correct object counts, while red borders denote counting errors. Prior baselines frequently fail to enforce correct counts or introduce visual artifacts, whereas ATHENA improves count fidelity while preserving scene structure and visual coherence, with the adaptive variant yielding the most consistent corrections.