ATHENA: Adaptive Test-Time Steering for Improving Count Fidelity in Diffusion Models

Mohammad Shahab Sepehri* Asal Mehradfar* Berk Tinaz
Salman Avestimehr Mahdi Soltanolkotabi
Department of Electrical and Computer Engineering
University of Southern California, Los Angeles, CA, USA
{sepehri, mehradfa, tinaz, avestime, soltanol}@usc.edu

Abstract

Text-to-image diffusion models achieve high visual fidelity but surprisingly exhibit systematic failures in numerical control when prompts specify explicit object counts. To address this limitation, we introduce ATHENA, a model-agnostic, test-time adaptive steering framework that improves object count fidelity without modifying model architectures or requiring retraining. ATHENA leverages intermediate representations during sampling to estimate object counts and applies count-aware noise corrections early in the denoising process, steering the generation trajectory before structural errors become difficult to revise. We present three progressively more advanced variants of ATHENA that trade additional computation for improved numerical accuracy, ranging from static prompt-based steering to dynamically adjusted count-aware control. Experiments on established benchmarks and a new visually and semantically complex dataset show that ATHENA consistently improves count fidelity, particularly at higher target counts, while maintaining favorable accuracy–runtime trade-offs across multiple diffusion backbones. Our code and data are publicly available at https://github.com/MShahabSepehri/ATHENA.

1. Introduction

Text-to-image (T2I) diffusion models have gained huge popularity and become the dominant paradigm for high-quality image generation, driven by their ability to synthesize images from natural language prompts with strong visual realism, semantic alignment, and compositional expressiveness [33].

Despite their strong generative capabilities, diffusion models continue to exhibit systematic failures in numerical control when prompts specify explicit object counts. 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"* often produce images with missing instances, duplicated objects, or ambiguous merges that obscure instance boundaries. While prompt sensitivity can influence generation outcomes and has motivated prompt optimization strategies [8], such interventions remain insufficient for reliably enforcing quantitative constraints [32]. As illustrated in Figure 1, these counting errors persist across multiple state-of-the-art T2I models, indicating a general limitation rather than a model-specific shortcoming.

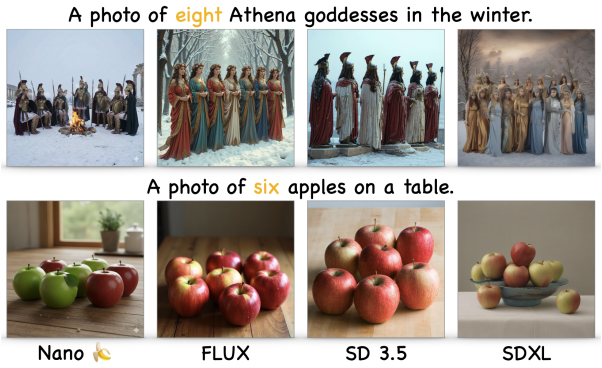


Figure 1. Count fidelity failures in T2I generative models.

Accurate object counting is important beyond visual plausibility. Downstream applications such as synthetic data generation for object detection and segmentation [29, 39] and simulation environments for robotics and embodied agents [28] rely on faithful instance-level control. In these settings, counting errors can propagate to downstream perception and decision-making, reducing the reliability of diffusion-based generation.

From a modeling perspective, object counting poses a natural challenge for diffusion-based generation. Diffusion models rely on a continuous denoising process without explicit mechanisms for enforcing discrete numerical constraints [13, 14]. Early stochastic decisions in the sam-

*Equal contribution.

pling trajectory largely determine object multiplicity and spatial structure, while later steps primarily refine local appearance [10, 23, 27, 30, 38]. As a result, counting errors become difficult to correct once they emerge, motivating intervention mechanisms that act before structural decisions become effectively irreversible [20].

Several recent works have explored mechanisms for improving counting fidelity in diffusion models, including guidance-based and iterative refinement approaches [2, 42]. While these methods improve counting performance under controlled settings, they often rely on model-specific assumptions or lack adaptive mechanisms to correct errors once the sampling trajectory deviates from the target count. As a result, reliable and efficient count control across diffusion backbones and realistic prompt distributions remains an open challenge.

In this work, we introduce ATHENA (Adaptive Trajectory Harmonization via Early Numerical Assessment), a test-time *adaptive* steering framework for improving object count fidelity in T2I diffusion models. ATHENA estimates object count from intermediate representations and uses this signal to modify the sampling trajectory before structural layouts become effectively fixed [5, 16]. The framework is model-agnostic, requires no retraining or architectural modifications, and generalizes across diverse diffusion backbones.

We instantiate ATHENA through three progressively adaptive steering variants that isolate the contribution of each control component. ATHENA-Static applies fixed steering, ATHENA-Feedback introduces count-aware conditional steering based on intermediate estimates, and ATHENA-Adaptive further adjusts steering strength online in response to persistent count errors. We evaluate these variants on CoCoCount [2], a balanced extension with a uniform count distribution (CoCoCount-E), and a newly constructed challenging benchmark (ATHENA dataset). Across datasets and diffusion backbones, ATHENA improves count fidelity while maintaining favorable accuracy–runtime trade-offs.

Contributions. Our main contributions are as follows:

- We propose ATHENA, a model-agnostic steering framework for improving object-count fidelity in T2I diffusion models that requires no architectural changes or retraining. ATHENA can be applied to arbitrary T2I models via lightweight test-time interventions and yields immediate improvements in numerical fidelity.
- We introduce the ATHENA dataset, a new evaluation benchmark that complements existing counting datasets by targeting challenging object categories (e.g., accordion) and compositional prompt structures with relational constraints (e.g., next to a river) and object-level distractions (e.g., with a person).
- Through extensive experiments on three datasets, we demonstrate that ATHENA improves count fidelity of the base diffusion models by up to 22% and outperforms baselines, while reducing memory usage by approximately $4\times$ and achieving up to $2.5\times$ faster image generation relative to the baselines.

2. Related Work

Prior work on T2I diffusion models has identified object counting as a challenge when prompts specify explicit cardinalities [8, 41]. *Make It Count* [2] improves counting through an auxiliary layout-modification model built on SDXL-specific internal features, limiting generality across diffusion backbones. *CountCluster* [20] instead enforces quantity control through early cross-attention clustering, but relies on spatial separation assumptions that become less reliable in cluttered scenes.

Several methods regulate object counts through gradient-based guidance or optimization during diffusion sampling. *Counting Guidance* [15] enforces target counts using gradients from a trained counting network, increasing inference cost and potentially disrupting global structure under large corrections. *YOLO-Count* [43] similarly applies guidance from dense cardinality predictions produced by a specialized counting model, but requires additional training and lacks adaptive error correction during sampling. *Detection-Driven Object Count Optimization* [42] instead performs iterative test-time refinement using detector feedback, resulting in slower inference and limited correction once early structural decisions become fixed.

Training-free and pipeline-based alternatives have also been explored for addressing counting errors. *CountDiffusion* [24] modifies attention maps using external count estimates from intermediate generations, but relies on task-specific auxiliary counters rather than adaptive steering. Beyond diffusion-level intervention, decomposition- or agent-based pipelines such as *MuLan* [22] generate objects sequentially or through sub-task planning, introducing additional computational overhead and deviating from standard diffusion sampling.

More broadly, guidance and steering techniques have been widely studied in diffusion models to improve conditional alignment. Attention-based interventions manipulate cross-attention to enhance semantic or spatial consistency [3, 10], while reward-based approaches refine image–text alignment using external signals [1]. However, these methods primarily target semantic fidelity or attribute binding rather than discrete global constraints such as exact object counts, particularly at higher cardinalities where early stochastic decisions dominate.

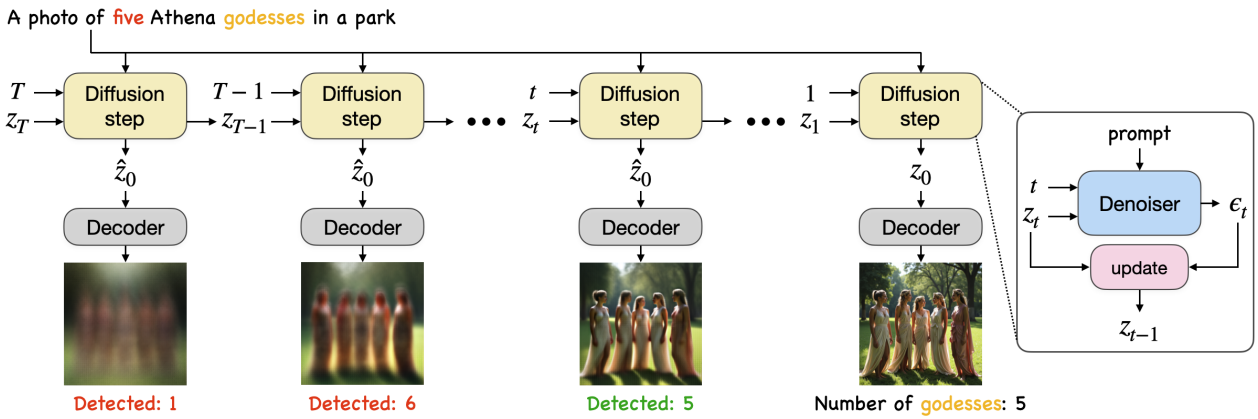


Figure 2. Count estimation across diffusion steps. Early decoded latents are too noisy for reliable counting, while later steps improve count estimation at higher inference cost. ATHENA performs count estimation at an intermediate diffusion step to balance reliability and efficiency.

3. Method

3.1. Problem Setup and Design Goals

We study test-time control of object count fidelity in diffusion-based T2I models. Let G_θ denote a pretrained generator that produces an image through an iterative sampling process, yielding a sequence of latent representations $\{z_t\}_{t=0}^{T}$ conditioned on a text prompt p specifying a target object count k for a particular object category. Our objective is to improve the accuracy with which the final generated image satisfies this target count. We assume access to intermediate latent or decoded representations during sampling, and all interventions are performed at test time without retraining or modifying the generator parameters. Additional diffusion preliminaries are provided in Sec. A.

ATHENA frames object counting as a discrete control problem over the diffusion sampling trajectory. It operates by monitoring intermediate generation signals and applying lightweight, test-time adjustments that influence the trajectory. These design goals define the class of adaptive, test-time interventions considered in the following sections.

3.2. Count Estimation Across Diffusion Steps

Diffusion-based T2I generation proceeds through a sequence of intermediate latent representations that transition from noise to a coherent image. At early sampling steps, object structure is insufficiently formed and decoded images are too noisy for reliable automated count estimation. At later steps, object instances are fully formed and count estimation becomes reliable, but this requires additional diffusion steps, increasing inference time. As illustrated in Figure 2, this tradeoff yields an intermediate regime in which object count can be estimated reliably without incurring the cost of late-stage sampling. Motivated by this observation, ATHENA performs count estimation at a fixed intermediate diffusion step t_{est}, decoding the corresponding clean latent estimate $\hat{z}_0^{(t_{\text{est}})}$ to balance estimation reliability and inference cost.

3.3. ATHENA: Test-Time Steering Framework

ATHENA is a test-time steering framework for improving object count fidelity in diffusion-based T2I generation, without modifying model parameters or requiring retraining. The framework estimates object count at an intermediate diffusion step and uses this signal to steer the sampling trajectory via prompt-based control, enabling corrective intervention before structural errors form. All components operate at inference time and are compatible with both deterministic and stochastic diffusion samplers.

3.3.1. Prompt-Based Steering Mechanism

At the core of ATHENA is a lightweight, training-free steering mechanism that modifies the denoising trajectory via prompt conditioning. At diffusion step t, the denoiser is evaluated twice on the current latent z_t: once with the original prompt p, producing $\epsilon_t \triangleq \epsilon_\theta(t, z_t, p)$, and once with a *control prompt* $\hat{p}$, producing $\hat{\epsilon}_t \triangleq \epsilon_\theta(t, z_t, \hat{p})$ (see Figure 3). These two predictions are combined to form a 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 (2)) is then applied using $\tilde{\epsilon}_t$ in place of ϵ_t to obtain z_{t-1}. We normalize $\tilde{\epsilon}_t$ to match the norm of ϵ_t, ensuring the steered latent remains within the denoiser's expected scale and preserves stable, in-distribution sampling.

The steering operation in (1) modifies the denoiser prediction using the difference term $(\epsilon_t - \hat{\epsilon}_t)$, which captures the change induced by replacing the original prompt p with the control prompt $\hat{p}$. By adding a scaled version of this difference to ϵ_t, ATHENA biases the sampling trajectory toward the target count while preserving semantic structure. Unlike classifier- or loss-based guidance, the steering term is computed entirely through forward denoiser evaluations

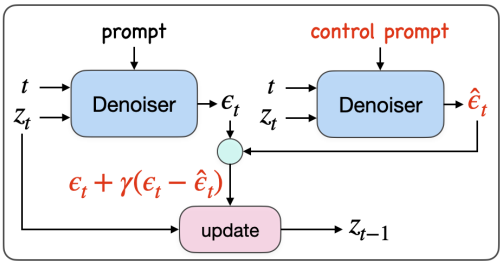


Figure 3. ATHENA steering block. The denoiser is evaluated with the original and control prompts to obtain ϵ_t and $\hat{\epsilon}_t$, which are combined to obtain the latent state z_{t-1} for test-time count control.

without backpropagation, making the framework computationally efficient and compatible with any architecture. The scalar γ controls the steering strength, interpolating between unsteered sampling ($\gamma = 0$) and stronger corrective influence.

3.3.2. Control Prompt Construction

ATHENA induces steering by evaluating the denoiser under two textual conditionings: the original prompt p and a modified *control prompt* $\hat{p}$, whose effect on the denoiser output is formalized in Equation (1). The control prompt provides an alternative conditioning signal that encodes corrective intent purely at the prompt level, while remaining fully compatible with the pretrained generator.

The framework imposes minimal assumptions on the form of $\hat{p}$. In practice, $\hat{p}$ is constructed by modifying only the object-count specification in the original prompt p, while keeping all other semantic content unchanged. We consider two forms of control prompts: (i) a *count-agnostic* prompt obtained by removing explicit cardinality constraints, and (ii) a *feedback-based* prompt in which the target count is replaced by an estimated count from an intermediate decoded generation. In all cases, $\hat{p}$ is treated identically to p by the underlying model and requires no architectural changes, auxiliary networks, or gradient-based optimization.

By separating control prompt construction from the steering mechanism, ATHENA decouples how corrective information is encoded from how it influences the sampling trajectory. Different choices of $\hat{p}$ yield distinct instantiations of the framework, while sharing the same denoiser evaluations and sampler interaction. We describe these instantiations in the following subsection.

3.3.3. ATHENA Control Strategies

We present three instantiations of ATHENA that progressively increase adaptivity while preserving the same test-time, training-free steering mechanism. All strategies rely on the prompt-based steering operation described in Sec. 3.3.1 and differ in whether and how intermediate generation signals are used to select the control prompt and steering

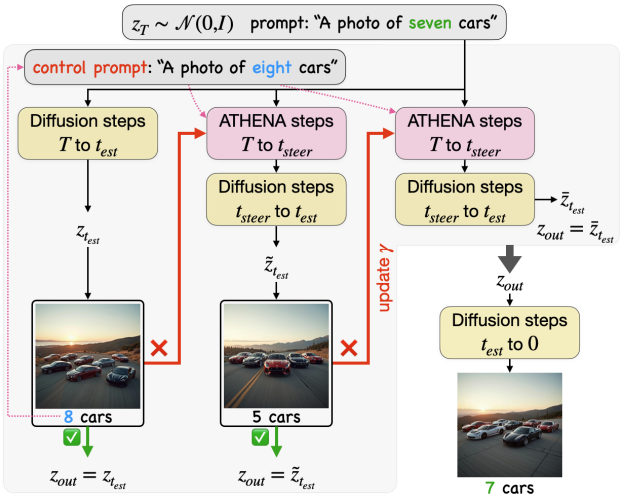


Figure 4. ATHENA-Adaptive pipeline. The method estimates object count at an intermediate diffusion step, applies early-stage prompt-based steering, adaptively adjusts the steering strength once based on the observed error direction, and completes generation using standard diffusion.

strength. This progression moves from fixed, count-agnostic control to feedback-informed and adaptive steering, enabling increasingly targeted correction of counting errors while maintaining a lightweight, model-agnostic design.

ATHENA-Static. ATHENA-Static applies prompt-based steering using a fixed, count-agnostic control prompt. Given an original prompt p specifying a target count k, the control prompt $\hat{p}$ is constructed by removing the explicit cardinality constraint while preserving all other semantic content. No intermediate count estimation is performed, and steering is applied once from the initial diffusion step to a cutoff t_{steer}, after which standard diffusion proceeds unmodified.

This variant isolates the effect of prompt-level steering without feedback or adaptive adjustment. It incurs minimal computational overhead and serves as a baseline demonstrating that count fidelity can be influenced through prompt-based steering. Details are provided in Sec. B.1.

ATHENA-Feedback. While static steering can improve count fidelity, its effectiveness depends on how well a fixed control prompt aligns with the trajectory. ATHENA-Feedback addresses this limitation by incorporating a single intermediate count estimate to inform the control prompt.

Specifically, diffusion is first run without steering until an intermediate step t_{est}, where the partially denoised latent is decoded and the object count is estimated. If the estimated count differs from the target, generation is restarted from the same initial noise using a feedback control prompt constructed by replacing the original count in p with the

Table 1. Structure of the ATHENA dataset with four levels of increasing prompt complexity.

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"

observed count. Prompt-based steering is then applied once during early diffusion steps down to a cutoff t_{steer}, after which diffusion proceeds unmodified to completion. No additional count checks or parameter updates are performed.

By correcting the semantic mismatch between the prompt and the emerging structure, ATHENA-Feedback improves robustness over static steering while remaining simple and efficient. Algorithmic details are provided in Sec. B.2.

ATHENA-Adaptive. Although feedback steering makes the control prompt feedback-informed, its success depends on the steering strength γ. If γ is too small, steering may reduce the counting error without reaching the target; if γ is too large, it can overshoot the target and flip the error direction, often at the expense of structural quality. ATHENA-Adaptive resolves this sensitivity through a single, direction-aware adjustment of γ based on intermediate feedback.

As illustrated in Figure 4, the method first estimates the baseline count at t_{est} without steering. If the estimate deviates from the target, prompt-based steering is applied once using the initial γ. If the resulting count still deviates from the target, the observed change indicates whether steering moved the generation toward or away from the target. When the error direction remains unchanged, γ is doubled; when it flips, γ is halved. A final steered generation is then performed using the adjusted γ, after which diffusion proceeds unmodified to completion.

Importantly, ATHENA-Adaptive does not perform iterative optimization or repeated parameter tuning. The error direction observed after a single feedback step provides a signal for adjusting the steering magnitude, striking a balance between corrective strength and structural stability. Further adjustment is unnecessary in practice, as additional iterations increase computational cost with diminishing returns. As a result, the method requires at most two steered trajectories, introduces no additional learned components, and improves robustness while preserving favorable accuracy–runtime trade-offs. Pseudocode is provided in Sec. B.3.

4. Experiments

We evaluate ATHENA across multiple T2I diffusion backbones and counting benchmarks to assess effectiveness, robustness, and computational trade-offs. Our experiments span three datasets of increasing complexity, including a newly introduced ATHENA dataset, and compare static, feedback, and adaptive steering against existing baselines. We report quantitative accuracy, analyze performance as a function of target count, and examine accuracy–runtime trade-offs alongside qualitative examples illustrating the impact of adaptive test-time steering.

4.1. Experimental Setup

Models. We evaluate ATHENA on three representative T2I diffusion backbones spanning different architectures and training regimes: SDXL [31, 36], SD 3.5 Large [7, 37], and FLUX.1-dev [18, 19]. All models are used as released, without modification or retraining.

Datasets. We conduct experiments on three counting benchmarks of increasing complexity. *CoCoCount* [2] consists of simple prompts with explicit numerical constraints and minimal scene context. To enable systematic analysis 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 dataset, a new benchmark designed to evaluate object counting under progressively challenging prompt conditions. The dataset consists of 360 prompts organized into four levels of increasing complexity, with target counts ranging from 2 to 10 (see Tab. 1). Prompts are generated using a large language model to systematically combine multiple constraints within a single instruction. The first level focuses on hard object categories (e.g., "accordion", "dumbbells"), while subsequent levels introduce additional challenges including scene context, distractor objects, and relational constraints. Additional construction details and statistics are provided in Sec. C.

Baselines. We compare ATHENA against unsteered diffusion sampling and representative prior methods for count-controlled image generation: *CountGen* [2] and *Counting Guidance* [15].

CountGen is specifically designed for SDXL and relies on model-specific components tied to that backbone. Accordingly, we evaluate CountGen on SDXL using the official implementation. Counting Guidance is not backbone-dependent, and its guidance formulation can be adapted to additional diffusion backbones with separate implementations.

Table 2. Quantitative 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 Dataset			
		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
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

The publicly released implementation is based on SD 1.4; to provide broader comparisons across modern diffusion backbones, we additionally implement Counting Guidance for FLUX.1-dev and SD 3.5 Large. We retain the original SD 1.4 implementation as an additional reference point, since its inference-time overhead is comparable to ATHENA-Adaptive on SDXL, enabling meaningful accuracy–runtime comparisons.

Although these baselines are not uniformly available across all backbones, they provide complementary comparisons for count-controlled generation across different models and guidance strategies.

Count Estimation. We estimate object counts using the pretrained open-vocabulary detector GroundingDINO [26]. The detector is used at test time both for intermediate count estimation (to guide feedback and adaptive steering) and for final evaluation of generated samples (see Sec. 4.2). It is neither fine-tuned nor integrated into the generative models. The same configuration is used across all methods and datasets, using the default hyperparameters provided by the model.

Our framework is agnostic to the choice of counting module: any estimator capable of providing intermediate count feedback (e.g., alternative detectors [4, 21, 25] or even human annotation) can be used in place of GroundingDINO. We use GroundingDINO as a representative open-source detector due to its strong accuracy and low computational overhead, with inference time negligible compared to the diffusion process. We further validate this design by replacing GroundingDINO with a large language model for intermediate estimation, as detailed in Sec. D.

Hyperparameter Selection. All ATHENA hyperparameters, including the estimation step t_{est}, steering horizon t_{steer}, and steering strength γ, are tuned exclusively on the CoCoCount dataset. Hyperparameter tuning is conducted separately for each diffusion backbone. Once selected, the parameters are fixed and reused without modification for CoCoCount-E and the ATHENA dataset.

We further evaluate the sensitivity of the adaptive update rule to the choice of multiplicative scaling factor (e.g., 1.5, 2, 4) used to increase or decrease γ, while all main results use a factor of 2. We observe that performance remains stable across the tested scaling factors, indicating that the adaptive mechanism is not sensitive to the exact update magnitude. Exact hyperparameter values and detailed sensitivity results are provided in Sec. E.

Computational Setup. All experiments are conducted using NVIDIA GPUs. We use NVIDIA RTX A6000 GPUs (48 GB) for CoCoCount and the ATHENA dataset, and NVIDIA A100 SXM4 GPUs (40 GB) for CoCoCount-E. For each dataset, the same GPU type is used across all models and methods to ensure fair runtime comparisons. Unless otherwise specified, all reported results use a fixed random seed. Absolute generation times may vary across datasets due to hardware differences; accordingly, we focus on relative accuracy–runtime trade-offs under fixed hardware.

4.2. Quantitative Results

Overall Counting Performance. Table 2 summarizes counting performance across diffusion backbones and datasets. ATHENA-Adaptive consistently improves exact-count accuracy over unsteered generation, with absolute gains of at least 11.8% (FLUX.1-dev on CoCoCount) and

up to 22.3% (SDXL on CoCoCount). Adaptive steering outperforms static and feedback variants in nearly all cases; the only exception is CoCoCount with FLUX.1-dev, where static steering performs slightly better. Overall, these results demonstrate that direction-aware adaptation of the steering strength is critical for robust count control across models and datasets. Seed sensitivity analysis (mean $\pm$ standard deviation over three seeds) is provided in Sec. F.1, showing that the observed performance gains are consistent across different random initializations.

Efficiency. Accuracy improvements are achieved with favorable efficiency trade-offs. As shown in Table 2 and Figure 5, ATHENA-Adaptive attains higher accuracy than CountGen while being over $2.5\times$ faster on the ATHENA dataset. At a comparable runtime to Counting Guidance on SD 1.4, ATHENA-Adaptive on SDXL achieves nearly $2\times$ higher counting accuracy. In terms of memory usage, CountGen requires 47.5 GB, Counting Guidance on SD 1.4 uses 17.2 GB, while ATHENA-Adaptive on SDXL requires only 11.9 GB. Thus, ATHENA achieves higher accuracy than prior baselines with approximately $4\times$ lower memory usage than CountGen, highlighting its efficiency as a lightweight, test-time control method.

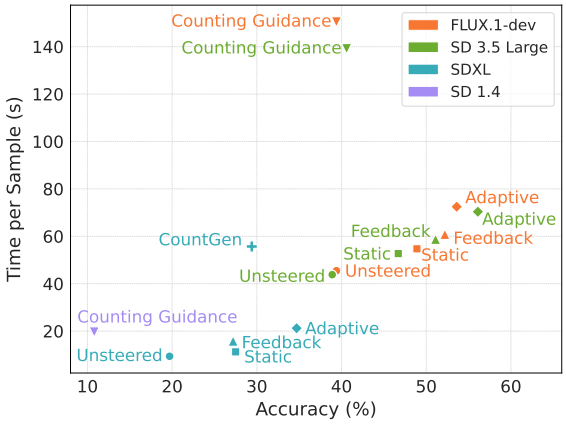


Figure 5. Accuracy–runtime trade-off on the ATHENA dataset. Colors denote the diffusion backbone, while marker shapes indicate the method.

We additionally analyze the number of extra diffusion steps introduced by each variant and find that ATHENA-Adaptive incurs only modest overhead, consistent with the runtime trends above. In practice, the average number of additional steps remains below a full diffusion pass across all evaluated models and datasets, while generations that already satisfy the target count usually incur no additional steps. Detailed statistics and additional accuracy–runtime figures are provided in Sec. F.2.

Robustness Across Target Counts. We analyze counting accuracy as a function of the target object count. Figure 6 reports results on CoCoCount-E with the SDXL backbone. Accuracy decreases as the target count increases for all methods, reflecting the growing difficulty of enforcing precise cardinality. Unsteered generation degrades sharply beyond three objects, and prior baselines exhibit limited robustness as counts increase. In contrast, ATHENA-Adaptive maintains consistently 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. These results indicate that adaptive steering mitigates early semantic drift and slows error accumulation as counting difficulty increases. Additional figures are provided in Sec. F.3.

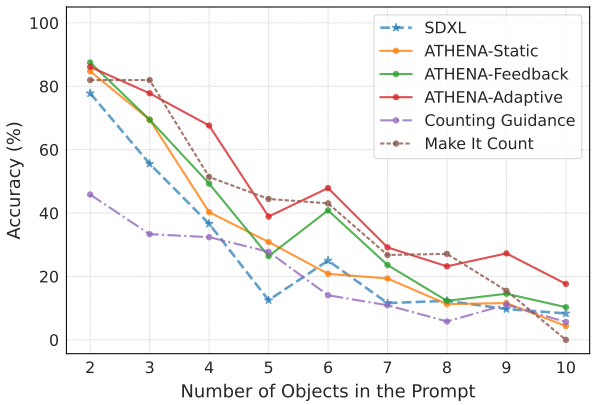


Figure 6. Counting accuracy versus target count on CoCoCount-E with SDXL backbone. ATHENA-Adaptive achieves high accuracy for small counts and nearly $2\times$ the accuracy of unsteered sampling at larger counts.

Evaluation on GenEval. To further validate the robustness of ATHENA under a standardized external benchmark, we evaluate ATHENA on the counting component of GenEval [9] using its default evaluation protocol and detector configuration. ATHENA improves accuracy over the unsteered baseline across all evaluated backbones; for example, on FLUX.1-dev, accuracy improves from 67.5% to 86.3%, while on SDXL it increases from 48.8% to 68.8%. These results indicate that the proposed steering mechanism generalizes across benchmarks and is not tied to the specifics of certain datasets. Detailed results, including runtime and additional error metrics, are provided in Sec. F.4.

Image Quality. We additionally evaluate the impact of test-time steering on visual quality using multiple perceptual metrics. We observe that ATHENA maintains comparable image quality to the unsteered baseline across all evaluated backbones, indicating that improvements in count fidelity do

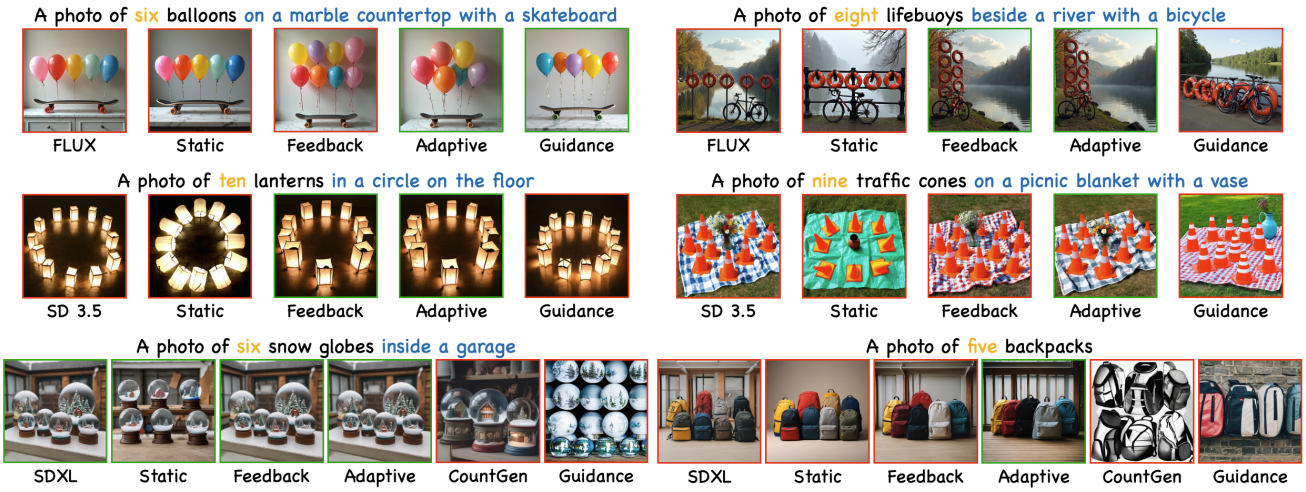


Figure 7. Qualitative results on complex, distractor-rich prompts. *Guidance* denotes Counting Guidance. Green borders indicate correct counts; red borders indicate incorrect counts. All ATHENA variants preserve structure and consistency while handling relational and multi-object instructions.

not come at the cost of visual quality. Detailed results are provided in Sec. F.5.

4.3. Qualitative Results

Figure 7 presents 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 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. Additional examples are provided in Sec. G.

5. Limitations

ATHENA focuses primarily on improving count fidelity for single-object counting prompts. While the proposed steering framework is potentially extendable to more complex compositional settings, the current formulation does not explicitly address prompts involving multiple object categories (e.g., *"three astronauts and two horses on a beach"*) or fine-grained attribute constraints (e.g., *"five Athena goddesses, two with closed eyes"*). In addition, Feedback and Adaptive variants introduce moderate inference-time overhead due to iterative steering, although the additional computation remains substantially smaller than several prior count-control approaches in our experiments. The effectiveness of the steering process can also be influenced by the quality of the intermediate feedback signal, although the framework itself remains compatible with different open-source or proprietary estimation modules. Extending the framework to richer semantic feedback signals and broader compositional control settings remains an important direction for future work.

6. Conclusion

We introduced ATHENA, a model-agnostic test-time steering framework for improving object count fidelity in T2I diffusion models without retraining or architectural changes. By estimating counts at an intermediate diffusion step and applying early prompt-based steering, ATHENA corrects numerical errors before structural mistakes become fixed. Across three diffusion backbones, ATHENA-Adaptive improves exact-count accuracy by up to 22.3%, achieves close to 80% accuracy for small target counts, and consistently outperforms prior baselines. These gains come with favorable efficiency: ATHENA is nearly $2.5\times$ faster than CountGen, achieves almost twice the accuracy of Counting Guidance at similar runtime, and requires roughly $4\times$ less memory on SDXL. Overall, ATHENA is an effective and efficient inference-time method for enforcing discrete numerical constraints in diffusion-based image generation while preserving visual quality.

Acknowledgements

We sincerely thank Willie Neiswanger and Justin Cho for their insightful feedback and valuable guidance. This work was partially supported by AWS credits through an Amazon Faculty Research Award, a NAIRR Pilot Award, and generous funding by Coefficient Giving. M. Soltanolkotabi and M. S. Sepehri were supported by the USC-Capital One Center for Responsible AI and Decision Making in Finance (CREDIF) Fellowship. M. Soltanolkotabi is also supported by the Packard Fellowship in Science and Engineering, a Sloan Research Fellowship in Mathematics, NSF CAREER Award #1846369, DARPA FastNICS program, NSF CIF Awards #1813877 and #2008443, and NIH Award DP2LM014564-01.

References

[1] Hyojin Bahng, Caroline Chan, Fredo Durand, and Phillip Isola. Cycle Consistency as Reward: Learning Image-Text Alignment without Human Preferences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 2

[2] 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*, pages 13242–13251, 2025. 2, 5, 15

[3] 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. 2

[4] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16901–16911, 2024. 6

[5] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11472–11481, 2022. 2

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 11

[7] 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. 5

[8] Haosheng Gan, Berk Tinaz, Mohammad Shahab Sepehri, Zalan Fabian, and Mahdi Soltanolkotabi. ConceptMix++: Leveling the Playing Field in Text-to-Image Benchmarking via Iterative Prompt Optimization. *Generative Models for Computer Vision Workshop @ CVPR*, 2025. 1, 2

[9] 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. 7, 21

[10] 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. 2

[11] 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*, pages 7514–7528, 2021. 22

[12] 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. 22

[13] Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1, 11

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 11

[15] 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)*, pages 899–908. IEEE, 2025. 2, 5

[16] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 2

[17] 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. 22

[18] Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024. 5

[19] 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. 5

[20] 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. 2

[21] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded Language-Image Pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10975, 2022. 6

[22] Sen Li, Ruochen Wang, Cho-Jui Hsieh, Minhao Cheng, and Tianyi Zhou. Mulan: Multimodal-llm agent for progressive and interactive multi-object diffusion. *arXiv preprint arXiv:2402.12741*, 2024. 2

[23] Shuangqi Li, Hieu Le, Jingyi Xu, and Mathieu Salzmann. Enhancing compositional text-to-image generation with reliable

random seeds. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[24] 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. 2

[25] Luping Liu, Zijian Zhang, Yi Ren, Rongjie Huang, Xiang Yin, and Zhou Zhao. Detector guidance for multi-object text-to-image generation. *arXiv preprint arXiv:2306.02236*, 2023. 6

[26] 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*, pages 38–55. Springer, 2024. 6

[27] Shweta Mahajan, Tanzila Rahman, Kwang Moo Yi, and Leonid Sigal. Prompting hard or hardly prompting: Prompt inversion for text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6808–6817, 2024. 2

[28] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021. 1

[29] Sergey I Nikolenko et al. *Synthetic data for deep learning*. Springer, 2021. 1

[30] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 23051–23061, 2023. 2

[31] 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. 5

[32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 1

[33] 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*, pages 10684–10695, 2022. 1

[34] Christoph Schuhmann. LAION Aesthetic Predictor. https://github.com/christophschuhmann/improved-aesthetic-predictor, 2022. 22

[35] 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. 11

[36] Stability AI. Stable Diffusion XL - Base 1.0. https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0, 2023. 5

[37] Stability AI. Stable Diffusion 3.5 - Large. https://huggingface.co/stabilityai/stable-diffusion-3.5-large, 2024. 5

[38] 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. 2

[39] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 969–977, 2018. 1

[40] 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. 22

[41] Xindi Wu, Dingli Yu, Yangsibo Huang, Olga Russakovsky, and Sanjeev Arora. Conceptmix: A compositional image generation benchmark with controllable difficulty. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. 2

[42] Oz Zafar, Lior Wolf, and Idan Schwartz. Detection-Driven Object Count Optimization for Text-to-Image Diffusion Models. *arXiv preprint arXiv:2408.11721*, 2025. 2

[43] 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*, pages 16765–16775, 2025. 2

A. Diffusion Preliminaries

A.1. Diffusion Models for Text-to-Image Generation

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,

$$z_{t-1} = \mathcal{S}_t(z_t, \epsilon_t, p), \tag{2}$$

where $\mathcal{S}_t(\cdot)$ may be stochastic or deterministic depending on the sampling scheme, encompassing diffusion samplers such as DDPM [14] and deterministic probability-flow variants [35] 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 sampler-defined reconstruction operator

$$\hat{z}_0^{(t)} = \mathcal{D}_t(z_t, \epsilon_t). \tag{3}$$

For the diffusion models evaluated in this work, this reconstruction is given by

$$\hat{z}_0^{(t)} = z_t - \sigma_t \epsilon_t, \tag{4}$$

where σ_t is a scheduler-dependent noise scale. The estimated clean latent may be decoded into image space for intermediate analysis when required.

A.2. Conditional Guidance and Sampling Dynamics

Conditional control in diffusion-based T2I models is commonly achieved via guidance, where the sampling trajectory is modified to satisfy a desired objective. Given a conditioning loss $\mathcal{L}(z_t, p)$, guidance-based methods adjust the denoiser output as

$$\hat{\epsilon}_t = \epsilon_t - s \, \nabla_{z_t} \mathcal{L}(z_t, p), \tag{5}$$

where $\epsilon_t \triangleq \epsilon_\theta(t, z_t, p)$ and s controls the guidance strength [6, 13].

Such guidance is effective for semantic or appearance-level control but is less suited for global structural properties, such as object count, which are largely determined during early sampling steps. This limitation motivates adaptive interventions that operate early in the sampling trajectory, before global structure becomes fixed.

B. Algorithmic Details of ATHENA

B.1. ATHENA-Static

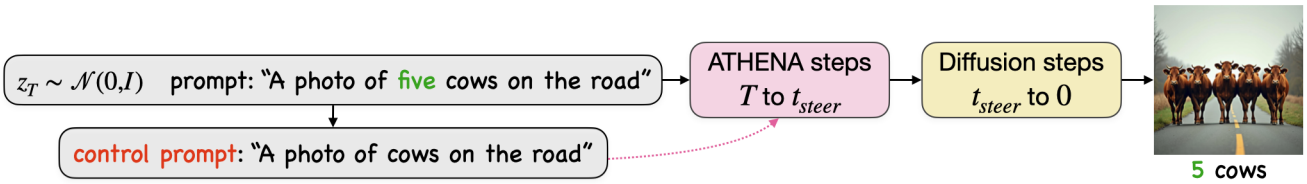


Figure 8. ATHENA-Static pipeline. Starting from the same initial noise, prompt-based steering is applied using a fixed, count-agnostic control prompt from diffusion step T down to a cutoff t_{steer}. Standard diffusion then proceeds unmodified to completion, yielding improved count fidelity without intermediate 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

B.2. ATHENA-Feedback

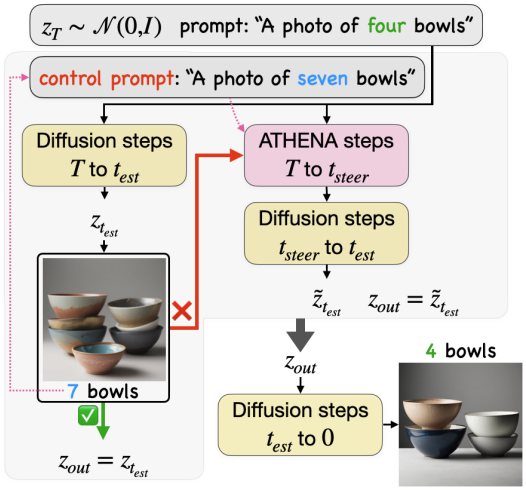


Figure 9. ATHENA-Feedback pipeline. An intermediate count is estimated at step t_{est} without steering. If the estimate differs from the target, generation is restarted from the same initial noise using a feedback control prompt, and a single early-stage steering phase is applied before completing diffusion.

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** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{est}})$
  $z_{out} = \tilde{z}_{t_{est}}$
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

B.3. ATHENA-Adaptive

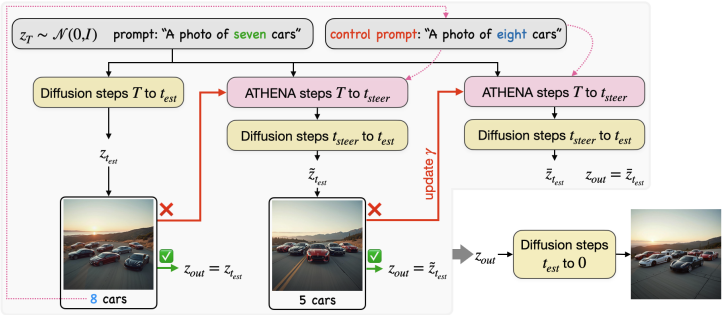


Figure 10. ATHENA-Adaptive pipeline. The method estimates object count at an intermediate diffusion step, applies early-stage prompt-based steering, adaptively adjusts the steering strength once based on the observed error direction, and completes generation using standard diffusion.

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
  $z_{t_{steer}} =$ **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** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{est}})$
  $\hat{z}_0 = \mathcal{D}_t(\tilde{z}_{t_{est}})$
  $\hat{x} = \text{Dec}(\hat{z}_0)$
  $c_2 = \text{Count}(\hat{x})$
  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** $(\epsilon_\theta, \mathcal{S}_t, p, t_s = t_{steer}, t_e = t_{est}, z_{t_{est}})$
    $z_{out} = \bar{z}_{t_{est}}$
  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

C. Dataset Construction and Statistics

C.1. CoCoCount and CoCoCount-E

The CoCoCount dataset [2] 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 prior work [2], we restrict the target count range to 2–10, as generation quality and counting accuracy degrade substantially beyond this range. We observe the same trend across all evaluated diffusion backbones.

Table 3. 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

C.2. ATHENA Dataset Generation

The ATHENA 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
```

C.3. ATHENA Dataset Details

The ATHENA dataset 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.

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."*

D. Alternative Count Estimation with GPT

To evaluate the dependence of our method on the choice of count estimator, we replace GroundingDINO 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 guide steering decisions.

In this experiment, only the intermediate count estimator is replaced, while final evaluation metrics are consistently computed using GroundingDINO across all methods. This isolates the effect of the estimator without altering the evaluation protocol. Results are summarized in Table 4.

Table 4. Effect of replacing the intermediate count estimator with GPT for the Feedback and Adaptive variants of ATHENA on CoCoCount and the ATHENA dataset.

Model	**Method**	**Estimator**	**CoCoCount**				**ATHENA Dataset**			
			Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	ATHENA-Feedback	GroundingDINO	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	GroundingDINO	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	GroundingDINO	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	GroundingDINO	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. This indicates that the effectiveness of the method does not depend critically on the specific choice of 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.

E. Hyperparameter Settings

E.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 5 summarizes the hyperparameter settings used for ATHENA across diffusion backbones and steering variants.

Table 5. 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

E.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 dataset are reported in Table 6.

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.

Table 6. Effect of the scaling factor in the adaptive update rule on the ATHENA 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

F. Additional Quantitative Results

F.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 7. 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. In particular, ATHENA-Adaptive maintains the highest accuracy across model–dataset pairs, while preserving favorable error metrics and runtime characteristics. 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 7. 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).

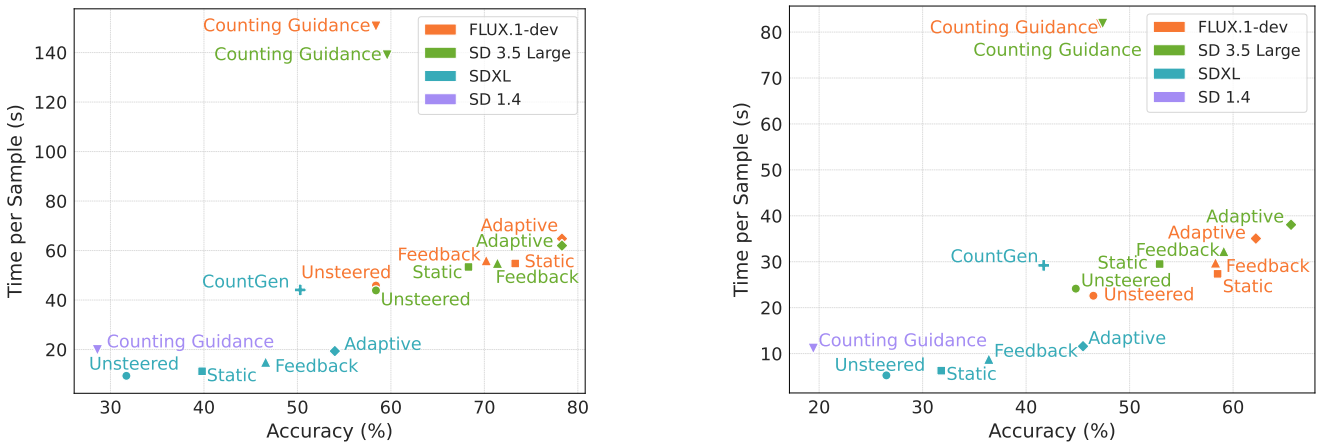
Model	Method	CoCoCount				CoCoCount-E				ATHENA Dataset			
		Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)	Acc (↑)	MAE (↓)	RMSE (↓)	Time (↓)
FLUX.1-dev	Unsteered	58.0±0.7	0.92±0.08	1.80±0.22	45.4±0.4	47.8±1.3	1.09±0.05	1.88±0.10	22.8±0.2	38.5±0.8	1.81±0.05	3.00±0.04	45.4±0.2
	ATHENA-Static	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

F.2. Efficiency

Additional accuracy–runtime comparisons across models and datasets are shown in Figure 11.

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.

Table 8 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



(a) CoCoCount (b) CoCoCount-E

Figure 11. Accuracy–runtime trade-offs on CoCoCount and CoCoCount-E. ATHENA-Adaptive consistently achieves higher accuracy at comparable or lower runtime than prior baselines.

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.

Table 8. Average number of additional diffusion steps introduced by each method.

Model	Method	CoCoCount	CoCoCount-E	ATHENA Dataset
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

F.3. Robustness Across Target Counts

Additional results analyzing counting accuracy as a function of the target object count are provided in Figure 12.

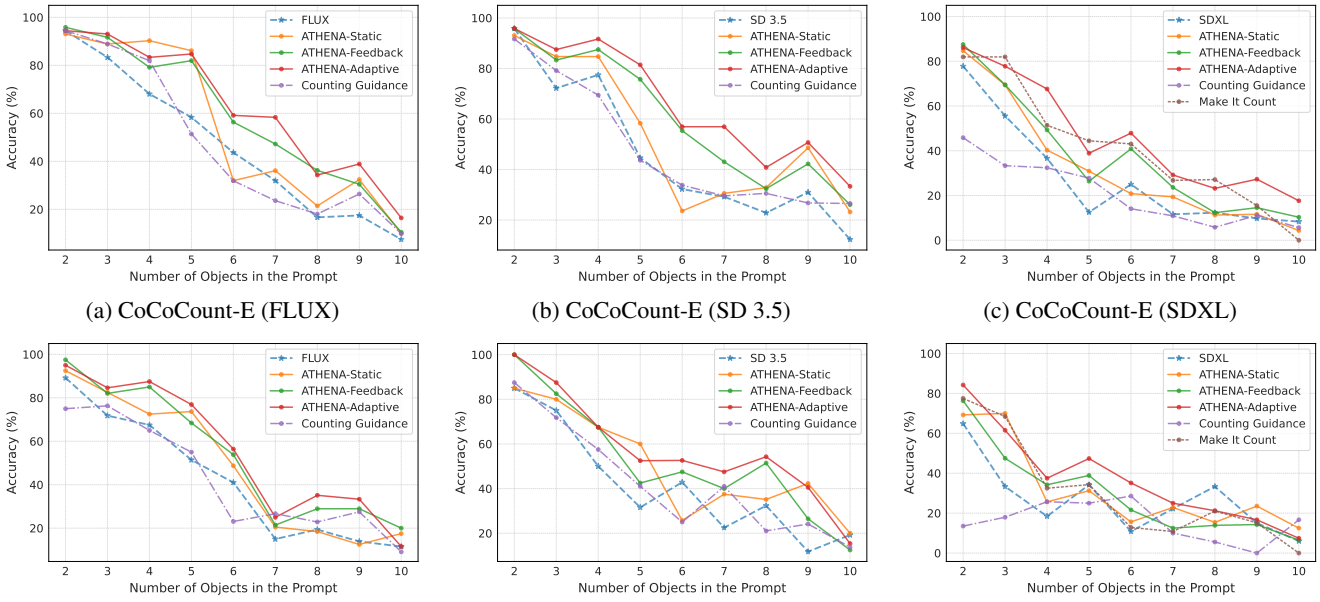
F.4. Evaluation on GenEval

We evaluate ATHENA on the GenEval benchmark [9], focusing on its counting component across the 80 prompts provided in the benchmark evaluation suite. GenEval provides a standardized detection-based evaluation protocol for compositional image generation, and we follow its default setup to ensure consistency with prior work. 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.

We observe that ATHENA consistently improves accuracy over the unsteered baseline across all evaluated backbones, and outperforms prior methods such as CountGen. Overall, these results demonstrate that the proposed steering mechanism improves count fidelity and generalizes to broader evaluation benchmarks.

F.5. 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.



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

(d) ATHENA dataset (FLUX) (e) ATHENA dataset (SD 3.5) (f) ATHENA dataset (SDXL)

Figure 12. 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.

Table 9. 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

We consider four widely-used reference-free metrics. CLIPScore [11] measures text-image alignment using CLIP embeddings. PickScore [17] and HPSv2 [40] are learned preference models trained to approximate human judgments of image quality. The LAION aesthetic score [34] 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 [12], 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 10. 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.

Table 10. 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 Dataset			
		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

G. Additional Qualitative Results

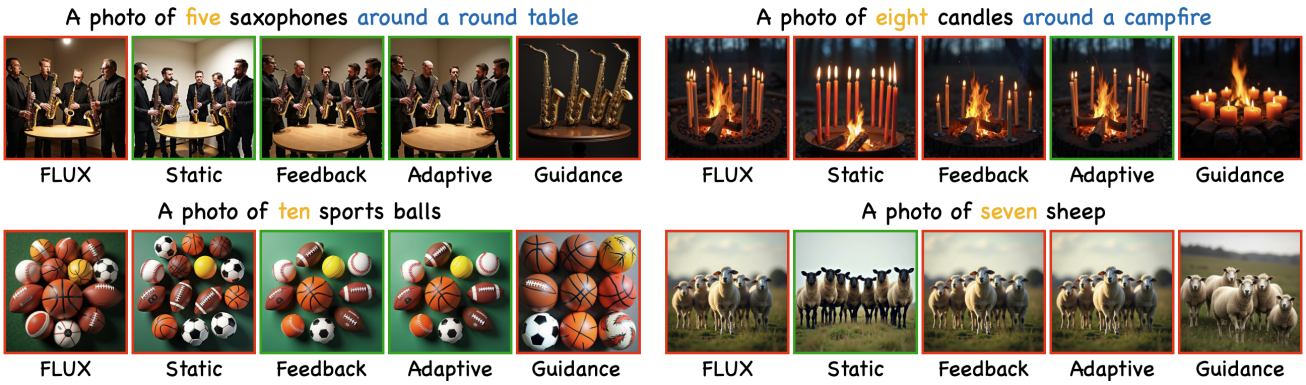


Figure 13. Additional qualitative results on relational and multi-object prompts. Results are shown for unsteered generation (FLUX) and the three ATHENA variants. Green borders indicate correct object counts, while red borders denote counting errors. ATHENA improves count fidelity across diverse settings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

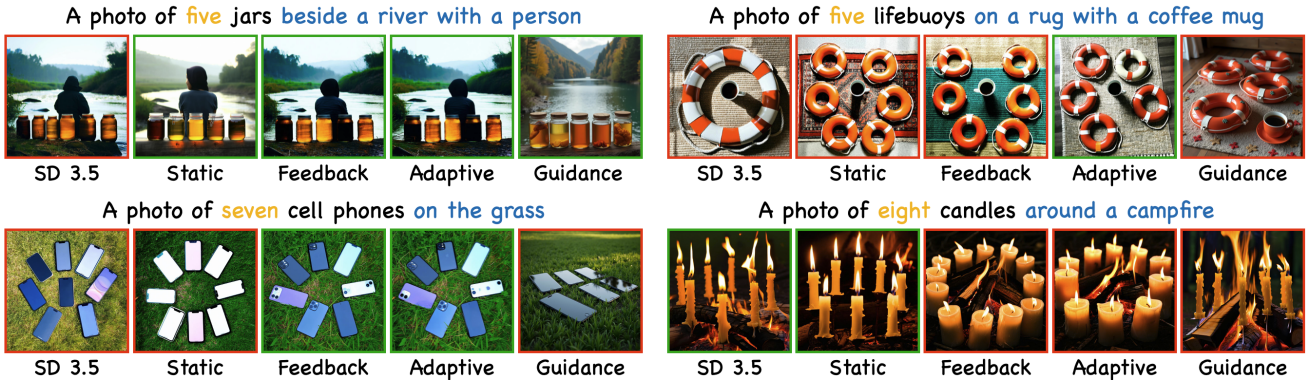


Figure 14. Additional qualitative results on relational and multi-object prompts. Results are shown for unsteered generation (SD 3.5) and the three ATHENA variants. Green borders indicate correct object counts, while red borders denote counting errors. ATHENA improves count fidelity across diverse settings (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 15. 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.