

COUNTLOOP: Iterative Agent Guided High Instance Image Generation

Anindya Mondal¹, Ayan Banerjee², Sauradip Nag³, Josep Lladós², Xiatian Zhu¹, Anjan Dutta¹

¹University of Surrey, ²Computer Vision Center, Universitat Autònoma de Barcelona,

³Simon Fraser University

Web — <https://mondalanindya.github.io/CountLoop/>

Abstract

Diffusion models have achieved remarkable progress in photorealistic synthesis, yet they remain unreliable for generating scenes with a precise number of object instances, especially in complex, high-density settings. We introduce COUNTLOOP, a training-free framework that equips diffusion models with accurate instance control via iterative structured feedback. It alternates between image synthesis and multimodal agent evaluation: an LLM-guided layout planner and critic provide explicit feedback on object counts, spatial arrangements, and attribute consistency, which is used to refine scene layouts and guide subsequent generations. Instance-driven attention masking and compositional techniques further prevent semantic leakage, enabling clear separation of individual objects even in occluded scenes. Evaluations on COCO-Count, T2I-CompBench, and two newly introduced high-instance benchmarks demonstrate that COUNTLOOP surpasses existing benchmarks by achieving a counting accuracy of as much as 98% while consistently acing spatial arrangement and visual quality over existing layout and gradient-guided baselines with a score of 0.97.

Introduction

Digital creators, designers, and artists increasingly use text-to-image diffusion models like DALL-E 3 (Betker et al. 2023), SDXL (Podell et al. 2024), and FLUX (Black-Forest-Labs 2024) to produce high-quality visuals. However, these models struggle with scenes containing many distinct yet related object instances, limiting their effectiveness in applications such as product advertising (*e.g.*, densely stocked retail shelves (Amazon Ads 2023; Team 2023)) or visualizing rare/extinct species (*e.g.*, dodos or mammoths (Yap 2024), where real data are limited; see Fig. 1). Current diffusion models typically saturate at around 10 instances per category (Binyamin, Tewel et al. 2024), yielding semantic drift (mixed attributes), spatial collapse (cluttered or overlapping objects), or instance duplication. For instance, a prompt like “31 cups on a coffee table” might produce only a dozen cups or an incoherent pile (Fig. 1), compromising both accuracy and usability.

Current solutions fall into two categories: (1) layout-first pipelines (Li et al. 2023; Feng et al. 2023; Liu et al. 2024;

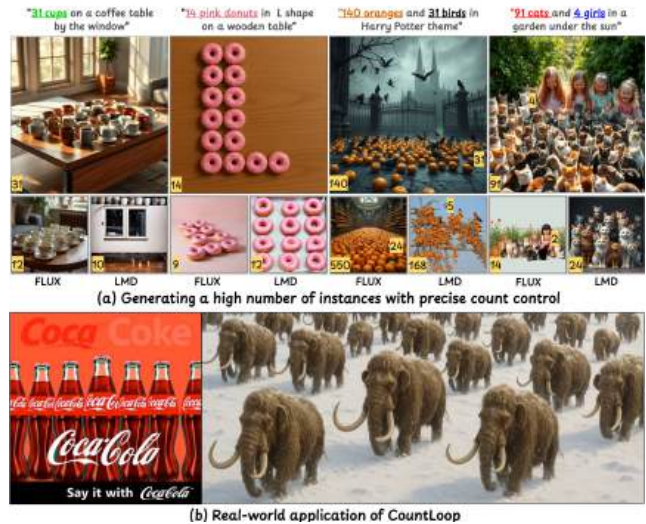


Figure 1: **COUNTLOOP** enables high-instance image generation with accurate layout and strong visual quality. (a) It handles scenes with 30-100+ instances, achieving precise counts and aesthetic layouts where prior models like FLUX and LMD struggle. (b) It supports real-world applications like product advertising and visuals of rare or extinct species.

Binyamin, Tewel et al. 2024; Zhou et al. 2024) and (2) gradient-guided methods (Chefer et al. 2023a). However, neither scale effectively to high-instance scenes or resolves the issues shown in Fig. 2. Layout-first pipelines use boxes or masks to guide diffusion, but often produce distorted semantics due to autoregressive biases in generative models (Xiong et al. 2024; Barron 2025) (*e.g.*, grid-like layouts that favor the top-left, see Fig. 2(a)). These approaches typically require annotated data or carefully engineered prompts (Binyamin, Tewel et al. 2024). Gradient-guided methods inject counting signals during denoising to steer generation towards the desired instance count, but they often introduce visual artifacts or worsen semantic leakage – an issue inherent to high-instance generation – especially as object density increases (Dahary et al. 2024a, 2025) (see Fig. 2(b)).

To address the persistent challenge of generating visually coherent scenes with precise object counts, we introduce COUNTLOOP, a framework that reimagines high-instance image generation as an iterative design process rather than a



Figure 2: Issues in High-Instance Image Generation

one-shot operation. Inspired by how human designers refine their work through successive iterations, COUNTLOOP establishes a closed-loop system where large language models serve dual roles: as creative planners that construct structured scene representations that capture object attributes and spatial relationships, and as critics evaluating the generated images against the original specification.

Crucially, COUNTLOOP integrates a cumulative attention mechanism during the denoising process to mitigate semantic leakage – a common issue in high-instance scenes. Rather than generating all subjects simultaneously, it provides per-instance grounding by preventing semantic entanglement and maintains the identity of individual objects. By imposing attention locality within instance-specific regions, COUNTLOOP encourages independence across objects and prevents the borrowing of features from nearby or similar instances.

This iterative agent-guided process involves generating an initial image, evaluating its alignment with the specification, and systematically refining the layout and prompt until quality thresholds are met. Together, these components enable COUNTLOOP to overcome the counting saturation and spatial coherence that affect existing approaches. Unlike prior methods requiring model retraining, COUNTLOOP acts as a plug-and-play enhancement to standard diffusion models, reliably producing dense scenes (100+ objects) with accurate instance counts and natural spatial distributions. Evaluations in COCO-Count, T2I-CompBench, and our new high-instance benchmark show that COUNTLOOP more than doubles counting accuracy while maintaining visual fidelity.

We summarize our contributions as follows: (1) We present COUNTLOOP, a training-free pipeline for generating high-instance images with precise object counts and strong aesthetic quality; (2) We introduce a planning graph to guide scene structure, combined with cumulative attention composition to prevent semantic leakage, even in dense scenes; (3) We develop an iterative procedure where a critic evaluates the generated image and provides feedback to refine the planning graph for the next iteration; (4) We conduct extensive evaluations on COCO-Count, T2I-CompBench, and new high-instance benchmarks, showing that our method achieves over 2x improvement in counting accuracy and significantly better visual coherence than existing baselines.

Related Work

Count control in Image Generation: Early diffusion models (LDM (Rombach et al. 2022), Imagen (Saharia et al. 2022), SDXL (Podell et al. 2024)) achieve stunning photorealism but falter beyond 10–15 identical objects, exhibiting attribute

leakage and spatial collapse (Chefer et al. 2023b,a; Dahary et al. 2024b,a). Layout-based approaches (Lian et al. 2023; Binyamin, Tewel et al. 2024) and gradient-guided corrections (Kang, Galim, and Koo 2023; Chefer et al. 2023a) offer partial fixes but demand heavy retraining or still fail in extremely dense scenes. COUNTLOOP avoids retraining by coupling a frozen diffusion backbone with an LLM-driven layout-refinement loop, incorporating a multi-turn object generation and composition mechanism that prevents the model from attribute leakage, thereby generating high-quality images even at high instance densities.

Layout-to-Image Generation: Techniques like GLIGEN (Li et al. 2023), LMD (Lian et al. 2023), and SceneLayoutNet (Zhang et al. 2024) use bounding boxes or masks for coarse count and placement control. They work well for moderate densities but degrade into rigid, grid-like layouts when scenes become crowded, lacking both relational reasoning and iterative correction. Scene-graph models (e.g., SG2IM (Johnson, Gupta, and Fei-Fei 2018)) capture pairwise relations but depend on extensive annotations, while LLM planners (e.g., LayoutGPT (Feng et al. 2023)) generate initial layouts that often violate natural groupings under high-instance prompts (see Fig. 2(a)). Recent studies (Dahary et al. 2024b,a) highlight the architectural tendency of attention layers to leak visual features between subjects – a phenomenon that complicates multi-subject generation. In our work, we propose a method to dynamically generate realistic looking layouts from prompt using an LLM and then process each layout independently to generate the corresponding object while preserving the texture in a iterative fashion. This prevents the model from sharing attention to similar looking objects in the image, thus preventing the attention leakage even during occlusions enabling the generation of accurate multi-instance object images without any extra optimization or training.

Agent-guided Diffusion Correction: Recent frameworks use LLMs as planners and critics to improve diffusion processes. They aim to improve outputs over time. For example, SLD (Wu, Lian et al. 2023) uses one LLM to find errors in generation and suggest changes to prompts. However, it treats the image as a black box and does not control the layout, which can lead to over-corrections that either repeat or leave out objects. GenArtist (Wang et al. 2024) uses multiple agents for tasks like editing color, style, and composition. It primarily focuses on improving aesthetics instead of counting instances or maintaining spatial relationships. RPG-DiffusionMaster (Yang et al. 2024) uses role-playing agents to draft and review prompts at different stages, enhancing narrative clarity. However, it does not address problems like object overlap or counting in dense or occluded scenes. While all three frameworks can improve prompts, they lack a clear representation of scenes. Consequently, their corrections may lack accuracy in dense scenes. In contrast, COUNTLOOP implements a targeted iterative refinement process specifically designed for high-instance generation. Our framework uses a structured planning graph representation that enables precise spatial reasoning between objects, coupled with a parameter-free textual optimizer that translates the LLM’s feedback into concrete layout modifications without altering model parameters and without any additional training.

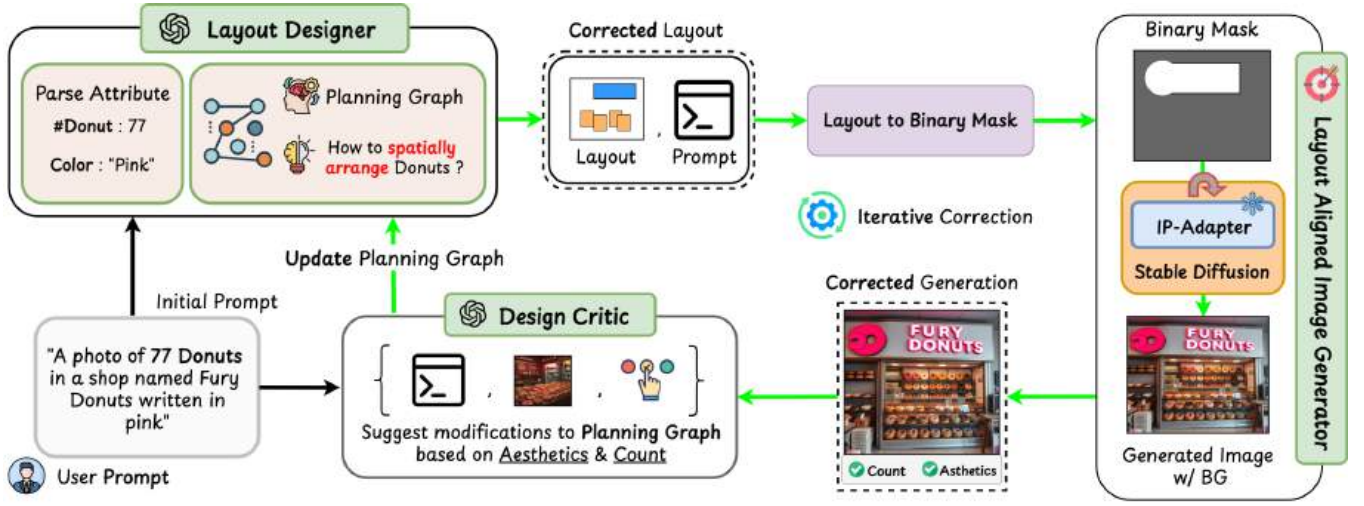


Figure 3: Given a text prompt, the Layout Designer constructs a planning graph encoding object attributes and spatial relations, which is converted into a pixel-aligned layout. Guided by instance masks and cumulative latent composition with an IP-Adapter, the image is synthesized. A Design-Critic evaluates the result and updates the planning graph via an iterative feedback loop. This loop repeats until the count and quality goals are met.

COUNTLOOP

We present COUNTLOOP, a training-free, LLM-guided approach for generating images with a high number of instances, ensuring precise object counts, coherent spatial arrangements, and distinct instance-level attributes as specified by a textual prompt (see Fig. 1).

Overview: We start by using a layout designer LLM to interpret the prompt and create realistic layouts that avoid rigid grid patterns (see Fig. 2(a)) while keeping natural object placements. These layouts guide style-consistent image generation using cumulative attention mechanism, which helps prevent attribute leakage (see Fig. 2(b)) and preserves object clarity, even when they overlap. Subsequently, a critic LLM assesses the generated output with respect to object count accuracy and overall aesthetic quality, offering structured feedback to iteratively refine both the layout and the prompt. This refinement loop continues until the output meets a set quality score, allowing us to generate complex images with many objects without retraining the diffusion model. Fig. 3 shows an overview of our method.

LLM-Guided Layout Generation

Generating images with precise control over multiple object instances – especially in dense scenes – remains challenging for text-to-image models, often resulting in unrealistic layouts or object overlaps. Layouts can be extracted from prompts via an LLM to guide image generation, with layout grounding (Lian et al., 2023) further enabling accurate object counting. However, due to limited spatial reasoning (Ramachandran et al. 2025) and autoregressive nature, LLMs often produce rigid, grid-like layouts (see Fig. 2(a)). To address this, we introduce structural reasoning into the LLM to promote more flexible arrangements. Inspired by scene graphs (Chen et al. 2024), we introduce planning graphs that refine LLM’s Chain-of-Thought reasoning by incorporating relational and spatial priors for layout generation. We call the resulting model

Layout Designer LLM, built on Qwen3 (Yang et al. 2025). This graph-based planning improves consistency in object placement, attributes, and relations, reducing grid artifacts and enabling more structured, realistic compositions.

Prompt parsing: As a precursor to our process, we break down the input prompt into its core components, which include object-level quantities, instance-level attributes, and instance-level quantities. For example, the prompt “two cats and a bird in the sky” contains two objects, “cat” and “bird”, with desired quantities of two and one, respectively. The object “bird” is associated with an instance-level attribute “in the sky”, which has a desired quantity of one, whereas the object “cat” is not associated with any instance-level attributes. We begin by instructing an LLM (Qwen3 (Yang et al. 2025)) to analyze the prompt and the attribute relations and return this information in a JSON dictionary. We guide the LLM with specific instructions on how to extract spatial relations from P as shown below.

Prompt Parsing Instruction

You are a scene planner. Given a prompt, return a JSON-based object-attribute relation with:

- **objects:** list of instance nodes with fields—id, category, position (x, y), depth, color
- **relations:** list of edges with fields—source, target, relation, distance, angle
- **context:** background scene type

These object-attribute relations serve as the foundation for the planning graph that injects spatial reasoning into the LLM’s chain-of-thought reasoning.

Planning Graph Construction: The graph construction process begins by utilizing object-attribute relations parsed from the input prompt. Specifically, the planning graph is defined as $G = (V, E, B_{bg})$, where V denotes object-instance nodes, E represents edges encoding spatial relations, and B_{bg} captures the scene context (e.g., “outdoor environment”). Each

node in V includes attributes such as category (e.g., cat, bird), a unique identifier (e.g., cat_1), normalized position $[x, y] \in [0, 1]^2$, depth prior $d \in [0, 1]$, and color. Edges in E encode spatial relations via directional operators (e.g., “above,” “left-of”), normalized distances, and angular orientations. The graph G enforces structured spatial reasoning, nodes specify individual properties while edges ensure relational consistency (e.g., minimum distances to prevent overlaps), enabling realistic multi-object scene construction. To integrate this structured representation into LLM reasoning, the graph is converted into a textual prompt template P_G :

$$P_G = \phi(['Object'], ['Relation'], ['Context']) \quad (1)$$

where ϕ denotes a text concatenation operator; ‘Object’ $\in V$, ‘Relation’ $\in E$, and ‘Context’ $\in B_{bg}$ denotes the textual attributes from the planning graph. Full prompt details are provided in the supplementary. The prompt P_G encodes object positions, depth, and sizes in text, enabling spatial reasoning within the LLM. This reasoning is combined with in-context examples for effective grounding:

Planning Graph Construction

Prompt: “A scene with 2 cats and 1 bird in the sky”

LLM Reasoning (Simplified):

1. Identify objects: 2 cats, 1 bird
2. Assign coarse positions: cats near center, bird above
3. Apply spatial jitter and avoid overlaps

Example: (Prompt: “A scene with 2 cats and 1 bird in the sky”) “objects”: [“id”: “cat 1”, “pos”: [0.3, 0.6], “d”: 0.4, “color”: “gray”, “id”: “cat 2”, “pos”: [0.6, 0.6], “d”: 0.4, “color”: “black”, “id”: “bird 1”, “pos”: [0.5, 0.3], “d”: 0.2, “color”: “white”], “relations”: [“from”: “cat 1”, “to”: “bird 1”, “r”: “below”, “dist”: 120, “angle”: 90], “context”: “outdoor, grassy field”

These examples provide a structured format that ensures precise object placement while preserving natural composition. Finally, both the planning graph prompt P_G and the in-context examples (denoted by P_{icl}) are fed into the Layout Designer LLM as follows:

$$\mathbb{J} = \text{LLM}(P_G, P_{icl}) \quad (2)$$

where \mathbb{J} is the LLM’s output in JSON format. From this, we extract the object layout coordinates \mathbb{L} , the scene description prompt P_d and background prompt P_{bg} respectively.

Layout Aligned Image Generation

After obtaining the layouts \mathbb{L} , our objective is to generate images that respect the layout. Layout grounded diffusion methods often suffer from attribute leakage in such scenarios (Dahary et al. 2024a, 2025), leading to correct object counts but compromised image quality (Fig. 2(b)). To address this, we draw inspiration from multi-turn image generation (Cheng et al. 2024), and instead of generating all object instances in a single pass, we adopt an iterative strategy – generating one instance at a time while preserving the texture using the previously generated content. This approach mitigates attention leakage into some other cases and ensures clearer separation of individual objects, even under occlusion.

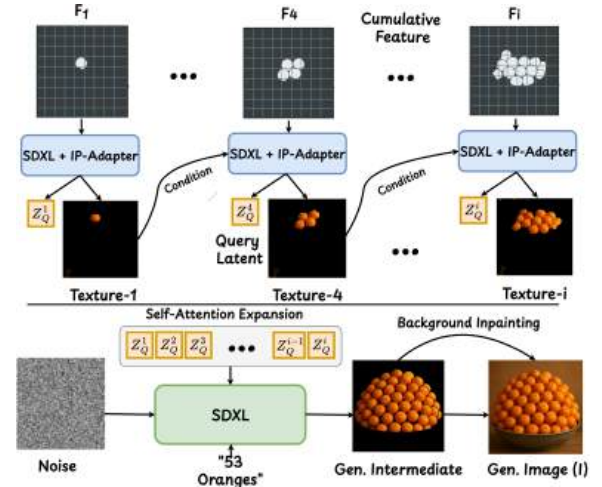


Figure 4: Cumulative Latent Composition along with self-attention expansion mitigates attribute leakage

Layout Aligned Attention Masking: Given the object layouts \mathbb{L} and prompt description P_d , we aim to ground the layout with the text to generate images with accurate instance counts. Since layouts are discrete spatial arrangements, we project them into a continuous space using a layout encoder. Specifically, we use the layout encoder of GLIGEN (Li et al. 2023), denoted by \mathbb{E} , which encodes each per-instance layout $l_i \in \mathbb{L}$ into latent embeddings $Q_i = \mathbb{E}(l_i)$. The full set of embedding is represented as $Q = \{Q_1, \dots, Q_N\}$. To ground these layout embeddings with the prompt P_d , we compute cross-attention A_{cross} , where the queries are layout embeddings Q , and the keys, values are derived from the text embedding of P_d . However, directly using A_{cross} for generation introduces semantic leakage as it attempts to generate all the instances at the same time. To mitigate this, we independently process A_{cross} at the instance level. For each object instance i , we apply a binary spatial mask $M_i \in \{0, 1\}^{w_i \times h_i}$ (1 inside the bounding box of l_i , 0 elsewhere), derived from the layout $l_i \in \mathbb{L}$. This mask is further refined via a self-segmentation algorithm (Dahary et al. 2024a) to obtain shape-aware masks. The masked layout feature is then computed as:

$$A_{mask}^i = A_{cross}^i \odot M_i \quad (3)$$

Here, A_{mask}^i denotes the instance-specific masked attention feature, which confines the receptive field of attention to the corresponding object’s region in the spatial domain.

Cumulative Latent Composition: Once instance-level attention maps A_{mask}^i are computed for each object layout $l_i \in \mathbb{L}$, we construct a coherent global latent feature map \mathbb{F} via cumulative composition in the diffusion latent space. Starting from a zero-initialized canvas, we iteratively paste each A_{mask}^i at its designated spatial location, producing intermediate latent maps $\mathbb{F}_i \in \mathbb{R}^{H_F \times W_F \times D}$, where H_F and W_F are spatial dimensions and D is the feature dimension. The composition is defined as:

$$\mathbb{F}_{i+1}(x, y) = \mathbb{1}_{(x, y) \in l_i} \cdot \text{Blend}(\mathbb{F}_i(x, y), A_{mask}^i)$$

Here, $\mathbb{1}$ indicates whether pixel (x, y) lies within the bounding box of l_i , and $\text{Blend}(\cdot)$ denotes feature concatenation.

This iterative process yields a sequence of cumulative latent feature maps $F = \{F_1, F_2, \dots, F_N\}$, where each F_i contains an increasing set of composed instances (see Fig. 4). When these disentangled instance-wise latent features are used for image generation independently, the cross-attention mechanism from Eq. 3 ensures per-instance grounding. This prevents semantic entanglement and maintains the identity of individual objects.

Appearance Consistency via IP-Adapter: Generating images independently from disentangled features F mitigates semantic leakage but often results in texture inconsistency, as each latent F_i undergoes separate noise and denoising processes. To address this, we condition the text-to-image diffusion model (e.g., SDXL (Podell, Liu et al. 2023)) on the foreground texture of the previously generated output using IP-Adapter (Ye et al. 2023). Since semantic leakage arises when query tokens attend to different instances during self-attention (Dahary et al. 2024a), we additionally preserve the per-instance query representation (Z_q) prior to its interaction with keys and values, thus maintaining instance-level semantics. This is formalized as:

$$I_{i+1}, Z_q^{i+1} = \Phi(F_{i+1}, P_d, \theta(I_i)) \quad \forall i \in \{1, \dots, N-1\} \quad (4)$$

Here, I_i is the image generated from F_i , N is the number of objects, and θ denotes IP-Adapter conditioning. The first image is generated without IP-Adapter due to the absence of a prior texture. We iterate over all F_i , enforcing consistency between the prompt P_d and prior visual cues to reduce hallucinations and preserve object distinctiveness. Once all clean query representations $Z_q = \{Z_q^1, Z_q^2, \dots, Z_q^N\}$ are extracted from the cumulative features F , we generate an image with semantically disentangled objects and minimal attribute leakage. To achieve this, we adapt the self-attention mechanism in standard diffusion models to a object-aware self-attention, analogous to attention expansion techniques in video diffusion (Wu et al. 2023; Alimohammadi et al. 2024), which promote temporal consistency. Similarly, our approach enables instances to attend to share the semantic features among each other. The attention is designed as follows:

$$\mathbb{A}([Z_q^1, \dots, Z_q^N], K, V) \quad (5)$$

where K and V denote keys and values (see Fig 4). This formulation ensures semantic coherence in the foreground, as each query Z_q^i interacts with a constant keys and values, while background features remain decoupled. Since the background is generated solely from P_d , it may lack realism. Thus, we inpaint the masked background using SDXL(Podell, Liu et al. 2023) conditioned on a background prompt P_{bg} . The final image I (see Fig. 4) captures the intended scene layout, with semantically disentangled foregrounds and minimal attribute leakage.

Layout Refinement via Iterative Feedback

After generating a layout-grounded image I , we ensure that the prompt description P_d is accurately reflected in terms of object count and image aesthetics. To this end, we propose an iterative refinement process that evaluates I , identifies flaws, and updates both the layout and prompt until the output meets the desired quality.

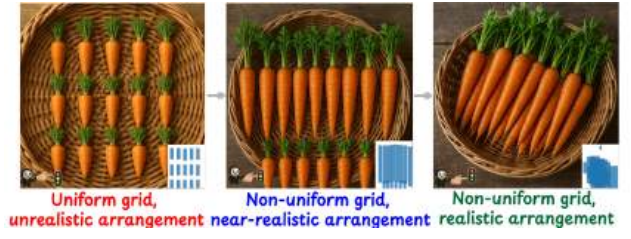


Figure 5: Successive Layout Refinement using LLM Critic. Corresponding layouts in the inset.

Design-Critic LLM: We reuse the LLM agent based on Qwen3 (Yang et al. 2025), repurposed as a Design Critic for analyzing generated images and suggesting prompt and layout corrections. Given a user prompt P_d and the corresponding generated image I , the agent evaluates two key aspects: (a) *object count accuracy* and (b) *visual aesthetics*, as illustrated in Fig. 3. Since LLMs may struggle to reliably assess object counts or aesthetic quality, we provide reference metrics to guide evaluation. Specifically, we estimate *count accuracy* using an open-vocabulary object detector (Liu et al. 2024), producing a score s_c . Additionally, we assess *visual quality* using an aesthetic scorer (Wu et al. 2024) that considers both P_d and I , outputting a score s_a . We define a composite score S to quantify overall image quality:

$$S = \alpha \cdot \max\left(0, 1 - \frac{|s_c - s_c^{gt}|}{s_c^{gt}}\right) + \beta s_a \quad (6)$$

where s_c^{gt} is the ground truth object count from the prompt, and $\alpha=0.6, \beta=0.4$ are weighting coefficients. This score balances object count accuracy and visual fidelity. The composite score S , along with I and P_d , is then fed to the LLM critic, which programmatically generates textual feedback, e.g., 'cat₁ is overlapping with cat₂', 'only 2 birds detected but target is 1', or 'lighting is inconsistent across objects'. This feedback is used to iteratively refine the spatial layout, aiming to improve both object count accuracy and visual quality.

Parameter-free refinement: The critic LLM's textual feedback must be translated into concrete edits to the planning graph to generate an updated image incorporating the feedback. Instead of fine-tuning model parameters—which is impractical without large annotated datasets—we introduce a gradient-free textual optimizer, ImGrad, inspired by TextGrad (Yuksekgonul et al. 2025). ImGrad acts as an intelligent text-editing agent, interpreting the critic's feedback ΔG to update the planning graph G :

$$G' = \text{ImGrad}(G, \Delta G),$$

Importantly, ImGrad operates on the textual representations rather than numerical parameters. For example: ① For feedback like "cup₇ is overlapping with cup₃", ImGrad adjusts G to increase spatial separation. ② For "only 28 cups detected but target is 30", it adds missing nodes in G . This parameter-free approach is compatible with any frozen diffusion model and allows precise, structured refinements. With

the updated graph G' , we re-run the planning graph construction, which produces a refined layout \mathbb{L} , followed by image synthesis I (see Fig. 5). The feedback loop terminates when the composite score S exceeds a threshold (0.85) and the predicted object count s_c matches the ground truth s_c^{gt} .

Experiments

Benchmarks: We evaluate COUNTLOOP on four datasets targeting image generation under varying instance counts and compositional complexity: **T2I-CompbenchCount:** A subset of T2I-Compbench (Huang et al. 2023), focused on open-world compositional text-to-image generation. **COCO-Count:** A subset of MS-COCO (Lin et al. 2014). **COUNTLOOP-S** (Single Category, High Instance): A custom dataset of 200 prompts, each describing a single category with 30–200 instances (e.g., “50 trees in a forest”). **COUNTLOOP-M** (Multiple Categories, High Instance): A custom dataset of 200 prompts, each with multiple categories, each ranging from 30–200 instances (e.g., “30 people and 20 cars in a city street”). More details about these newly introduced benchmarks are given in the supplementary material.

Evaluation Metrics: We evaluate object counting using GroundingDINO (Liu et al. 2024), chosen for its strong open-vocabulary detection performance across diverse categories. We report both F1 score and counting accuracy. For aesthetic quality and semantic alignment, we use the CLIP-FlanT5 encoder from VQAScore (Li et al. 2024), which provides scores between 0 and 1. Additionally, we conduct a human evaluation to assess how well the generated images match the prompts and to rate their overall visual appeal.



Figure 6: COUNTLOOP maintains precise object counts and natural arrangements in dense scenes, while baselines exhibit abnormal counts, spatial collapse, and grid artifacts.

Baselines: We compare COUNTLOOP against eleven representative baselines across three categories: text-to-image (T2I), agentic, and layout-to-image (L2I) approaches. **(1) T2I Models:** *SDXL* (Podell et al. 2024), a high-resolution diffusion model; *FLUX* (Black-Forest-Labs 2024), known for enhanced style and texture fidelity; *SD 3.5* (Stability-AI 2025), the latest version of Stable Diffusion; *Counting Guidance* (Kang, Galim, and Koo 2023), which integrates a counting module into denoising; and *GPT-4o* (Yan et al. 2025), a proprietary multimodal model capable of image synthesis. **(2) Agentic Models:** *GenArtist* (Wang et al. 2024), an agent-based framework for iterative artistic refinement; *SLD* (Wu, Lian et al. 2023), a self-correcting loop for accuracy enhancement; and *RPG-DiffusionMaster* (Yang et al. 2024), which uses role-playing agents for multi-stage generation. **(3) L2I Models:** *LMD* (Lian et al. 2023), leveraging LLM-generated layouts; *MIGC* (Zhou et al. 2024), focused on multi-instance layout-constrained synthesis; and *CountGen* (Binyamin, Tewel et al. 2024), which uses attention manipulation for count control.

Quantitative Results: Table 1 presents a quantitative comparison across four benchmarks: COCO-Count, T2I-Compbench, COUNTLOOP-S, and COUNTLOOP-M. COUNTLOOP consistently outperforms baselines in both counting accuracy and spatial coherence across all categories. On COCO-Count and T2I-Compbench (moderate instance counts), COUNTLOOP achieves F1 scores of 98.47% and 95.38%, surpassing agentic methods like SLD (90.34% and 91.50%) and RPG-DiffusionMaster (84.89% and 91.32%), as well as T2I models such as GPT-4o (92.91% and 94.19%) and FLUX (84.73% and 90.75%). For high-instance datasets COUNTLOOP-S and COUNTLOOP-M (30–200 instances), COUNTLOOP attains F1 scores of 60.00% and 85.43%, clearly exceeding L2I baselines like MIGC (54.16% and 81.06%) and agentic approaches such as GenArtist (51.00% and 77.87%). These results highlight the challenges faced by competing methods under high-instance conditions, where agentic models outperform T2I but still fall short in precision. Additionally, COUNTLOOP maintains superior spatial coherence (0.73–0.97), affirming its capacity for accurate counting and spatial arrangement in complex scenes.

Qualitative Results: Fig. 6 demonstrates COUNTLOOP’s consistent precision across diverse instance counts. For “17 vases”, baselines under generate (LMD: 13, Count Guidance: 9, CountGen: 6), while COUNTLOOP accurately renders all 17 with natural arrangements. In the “104 hot air balloons” scene, COUNTLOOP precisely places all balloons with realistic spacing, unlike Count Guidance (57), CountGen (54), and LMD’s artificial clusters (225 overlapping). For dense scenes like “49 oranges in a bowl”, COUNTLOOP maintains exact count with natural overlaps and lighting, outperforming Count Guidance (55) and CountGen (22). Similarly, COUNTLOOP correctly generates 77 distinct donuts with varied textures, while LMD produces only 10 and Count Guidance yields 47. Crucially, COUNTLOOP consistently avoids semantic drift, grid artifacts, and count inaccuracies that outperforms baselines for high-instance image generation.

Ablation and Analysis: We perform a systematic ablation study on the COCO-Count dataset to evaluate the impact

Table 1: **Comparing counting and aesthetic quality across four benchmarks.** For every dataset we report *Counting*—split into **F1** score and **Accuracy**—and *Spatial*, which is the aesthetic quality.

Model		Single Category									Multi Categories		
		COCO-Count			T2I-Compbench			COUNTLOOP-S			COUNTLOOP-M		
		Counting (%) F1	Acc.	Spatial ↑	Counting (%) F1	Acc.	Spatial ↑	Counting (%) F1	Acc.	Spatial ↑	Counting (%) F1	Acc.	Spatial ↑
T2I	SDXL (Podell et al. 2024)	71.87	42.13	0.38	84.36	44.00	0.75	55.40	24.49	0.63	77.84	67.25	0.55
	FLUX (Black-Forest-Labs 2024)	84.73	54.19	0.53	90.75	57.00	0.78	49.08	29.59	0.65	79.99	78.00	0.58
	SD 3.5 (Stability-AI 2025)	83.97	50.56	0.46	88.56	50.00	0.76	54.96	33.67	0.64	79.91	77.19	0.56
	Counting Guidance ((Kang, Galim, and Koo 2023))	67.54	18.50	0.63	71.41	17.50	0.56	36.67	10.20	0.47	64.42	25.90	0.41
	GPT-4o (Yan et al. 2025)	92.91	72.50	0.55	94.19	68.00	0.80	49.45	39.64	0.69	79.10	50.11	0.60
Agentic	GenArtist (Wang et al. 2024)	75.40	45.50	0.45	85.33	55.82	0.70	51.00	30.56	0.60	77.87	70.34	0.57
	SLD (Wu, Lian et al. 2023)	90.34	69.90	0.70	91.50	65.50	0.77	55.04	40.07	0.75	82.46	74.35	0.65
	RPG-DiffusionMaster (Yang et al. 2024)	84.89	60.73	0.60	91.32	60.00	0.75	51.89	34.38	0.70	80.16	71.46	0.62
L2I	LMD (Lian et al. 2023)	54.69	29.81	0.24	71.44	35.50	0.73	49.24	28.57	0.66	80.28	77.67	0.64
	MIGC (Zhou et al. 2024)	73.82	36.11	0.36	71.47	33.00	0.65	54.16	25.17	0.65	81.06	79.08	0.62
	CountGen (Binyamin et al. 2024)	58.99	50.00	0.61	63.75	19.78	0.75	48.18	41.40	0.72	72.00	45.33	0.69
	COUNTLOOP (ours)	98.47	93.33	0.93	95.38	78.50	0.79	60.00	55.00	0.97	85.43	83.67	0.73

of key components in our model. Specifically, we ablate the *Planning Graph* and *Cumulative Attention* modules. As shown in Table 2, incorporating spatial reasoning through the planning graph enhances generation quality with complex reasoning, which is visually evident in Fig. 7. The cumulative attention module improves visual fidelity by $\sim 18\%$, highlighting its role in mitigating attention leakage in high-instance scenarios. Further analysis of the number of iterations required for optimal aesthetics (see supplementary material) reveals that two iterations are sufficient. Additionally, we plot count accuracy against the number of instances (see supplementary material) and demonstrate that our model maintains high accuracy, even when generating around 100 instances. We also compare the runtime of our best-performing model in terms of both count and aesthetics with existing approaches, reporting the time required to generate images with correct count and aesthetics in the supplementary material. Finally, we evaluate the performance of COUNTLOOP using different LLMs and image generation models, with results provided in the supplementary material.

Table 2: Ablation of design components

Planning Graph	Cumulative Attention	Metrics	
		Accuracy ↑	Spatial ↑
✗	✗	65.8	0.58
✓	✗	77.3	0.70
✗	✓	80.4	0.76
✓	✓	93.33	0.93

Human Evaluation: We evaluate COUNTLOOP with 30 participants (20 designers, 10 AI artists) recruited from design studios, freelance platforms, and AI art communities. Participants rated 15 sets of 3 blinded images per method – COUNTLOOP, LMD (Lian et al. 2023), and GPT-4o (Hurst et al. 2024) – from the COCO-Count dataset on a 5-point scale for Prompt Alignment, Aesthetic Quality, and Overall Preference. COUNTLOOP outperformed baselines ($p < 0.01$), scoring 4.6, 4.7, and 4.5, respectively (see Table 3). The users generally appreciated COUNTLOOP for the image generation quality being on par with GPT-4o in addition to its precision in instance generation. We discuss more details on how we

collected the data, who responded, and a screenshot of the Google Form used to collect responses in the supplementary.

Table 3: User Evaluation (5 best, 0 worst).

Metric	COUNTLOOP	LMD	GPT-4o
Prompt Alignment	4.6	3.8	4.0
Aesthetic Quality	4.7	3.9	4.1
Overall Preference	4.5	3.7	3.9



Figure 7: Spatial reasoning in image generation. Vanilla LLM (LMD (Lian et al. 2023)) fails to identify directions

Conclusion

We presented COUNTLOOP, a training-free, iterative framework that enables high-instance image generation with precise object counts and strong visual quality. By combining LLM-based planning graphs, instance-driven attention, and cumulative latent composition, COUNTLOOP overcomes key limitations of existing methods – such as count saturation, semantic leakage, and rigid layouts. A critic-in-the-loop further refines generation through layout and prompt updates. Evaluations on COCO-Count, T2I-CompBench, and a new high-instance benchmark show COUNTLOOP achieves over $2\times$ improvement in counting accuracy while preserving aesthetics, scaling reliably up to 100+ instances per image.

Limitations: While effective, COUNTLOOP inherits biases from pre-trained LLMs, struggles with view-grounded prompts and dense human scenes, and suffers from higher runtime due to its iterative nature.

Future Work: It would be interesting to extend COUNTLOOP toward layout-free generation with weak spatial pri-

ors, enhance human modeling in dense scenes, and enable ultra-high object counts through controllable upscaling or multi-canvas fusion.

Implementation Details

All experiments were conducted on a single NVIDIA A100 GPU (80GB) running Ubuntu 22.04, with Python 3.10, PyTorch 2.1, and CUDA 12.2. We used Stable Diffusion XL (sdxl-base-1.0) as the backbone diffusion model, configured with 50 denoising steps. Layout conditioning was implemented via the GLIGEN layout encoder (box+text mode), and cross-instance texture consistency was enforced using the IP-Adapter (public checkpoint from (Ye et al. 2023)). Both the Layout Designer and Design Critic agents were instantiated from the *Qwen* (Yang et al. 2025) large language model. Images were generated at a resolution of 1024×1024 , with composite score weights set to $\alpha = 0.6$ for count accuracy and $\beta = 0.4$ for aesthetic quality, a count detector confidence threshold of 0.3, and loop termination occurring when the composite score $S \geq 0.85$. A fixed random seed of 42 was used for all runs, and all third-party models and detectors were loaded from publicly released checkpoints. The workflow of our model is provided in Algorithm 1.

Benchmarks and Evaluation Details

Here we provide the details of the evaluation metric and the benchmark dataset used to judge the performance of our COUNTLOOP model.

GDINO as Counting Metric: Evaluating object counts in high-instance synthetic images is critical to measuring generative fidelity. While YOLO (Feng, Miao, and Zheng 2024) is widely used for real-time detection and counting, it is fundamentally limited by its grid-based architecture: YOLO often struggles in scenes with high object density, heavy overlap, or occlusion, leading to missed detections and duplicate counts. Moreover, YOLO cannot detect new object categories without retraining, restricting its use in open-set or zero-shot benchmarks. Grounding DINO (GDINO) (Liu et al. 2024) overcomes these limitations through a transformer-based design with denoising training and flexible query support. GDINO offers substantially higher precision and recall for counting in crowded scenes, handles both small and overlapping objects robustly, and can perform open-vocabulary detection – enabling evaluation on novel categories without additional training. In our experiments, GDINO delivers more reliable and consistent counts than YOLO, especially for our most challenging high-instance compositions a finding also claimed by CountGen (Binyamin, Tewel et al. 2024), where they had to use human evaluation for counting T2I Compbench images rather than YOLO due to disjoint categories. As shown in Fig. 8, GDINO’s robust spatial reasoning and zero-shot adaptability make it uniquely well-suited for high-fidelity counting in the settings addressed by our method.

COUNTLOOP-S & COUNTLOOP-M Benchmarks: Existing text-to-image (T2I) counting benchmarks, including T2I-Compbench (Huang et al. 2023) and COCO-Count (Binyamin, Tewel et al. 2024), suffer from several key limitations: (i) *Limited class diversity*—COCO-Count,

ALGORITHM 1: COUNTLOOP: Iterative Feedback for Precise Object Count and Aesthetics

```

Function COUNTLOOP (Prompt  $P$ , target counts  $n_{gt}$  // dict of
per-category counts
):
    /* Step 1: Parse prompt into planning graph */
     $G \leftarrow \text{BuildPlanningGraph}(P)$  // Construct
     $G = (V, E, B_{bg})$  with objects, relations,
    attributes (PAGE3)
    ;  $iter \leftarrow 0$ 
    while  $iter < \text{MaxIter}$  do
        /* Step 2: Generate layout and detailed
        prompts from planning graph */
         $(L, P_d, P_{bg}) \leftarrow \text{GenerateLayout}(G)$  // Pixel-aligned
        layout  $L$ ;  $P_d$  is the foreground prompt and
         $P_{bg}$  contains the background context
        (PAGE3-4)
        ;
        /* Step 3: Generate image using diffusion
        model (instance-by-instance) */
         $I \leftarrow \text{GenerateImage}(L, P_d, P_{bg})$  // Instance-aware
        denoising with attention masking,
        IP-Adapter for consistency, inpainting
        with background prompt (PAGE4-5)
        ;
        /* Step 4: Scoring */
         $s_c \leftarrow \text{CountScore}(I)$  // Per-category counts via
        GroundingDINO (PAGE6); dict matching  $n_{gt}$ 
        keys
        ;  $s_a \leftarrow \text{AestheticScore}(I, P_d)$  // CLIP-like
        aesthetic/alignment score (0-1) (PAGE6)
        ;  $S \leftarrow \text{CompositeScore}(s_c, s_a, n_{gt})$  // Combines
        normalized count accuracy and aesthetics;
        exact formula tuned empirically
        ;
        /* Step 5: Check stopping criteria */
        if  $S > 0.85$  and  $s_c = n_{gt}$  // Exact match across
        all categories
        then
            return  $I$ 
        /* Step 6: Critic feedback and refinement
        */
         $\Delta G \leftarrow \text{CriticFeedback}(I, P_d, s_c, s_a)$  // Structured
        LLM feedback on counts, spatial,
        attributes (PAGE3)
        ;  $G \leftarrow \text{ImGrad}(G, \Delta G)$  // Update planning graph
        ;
         $iter \leftarrow iter + 1$ 
    return  $I$ 

```

for example, samples only 20 classes from MSCOCO, excluding many real-world object types; (ii) *Restricted count range* – most benchmarks evaluate generation only for low-count scenes (typically < 10 objects), failing to challenge models on dense or high-instance compositions; and (iii) *Lack of complex multi-category prompts* – existing datasets rarely assess the ability to control multiple object types and their relationships within a scene. These constraints make it difficult to assess compositional and numeracy capabilities in state-of-the-art T2I systems rigorously.



Figure 8: Choice of GDINO accuracy as a counting metric over YOLOv9

To address these gaps, we introduce 2 new benchmarks: **COUNTLOOP-S** and **COUNTLOOP-M**. Both are constructed from 92 diverse classes curated from the OmniCount-191 dataset (Mondal et al. 2025). **COUNTLOOP-S** is designed for single-category, high-count evaluation (e.g., “A photo of 127 watches”), while **COUNTLOOP-M** targets multi-category control (e.g., “A photo of 148 birds and 6 dogs”), enabling assessment of compositional fidelity at scale. Representative generations are shown in Fig. 14; further qualitative examples are provided below.

Key Features:

- **High class diversity:** 92 categories, including *airplanes, apples, balloons, bananas, bears, birds, bowls, buttons, butterflies, cars, cats, dogs, donuts, elephants, fish, hot air balloons, laptops, monkeys, oranges, pineapples, rabbits, roses, sheep, suitcases, swans, teacups, tigers, trucks, turtles, vases, watches, wine glasses*, and more.
- **Broad count range:** Instance counts from 1 up to 100 and select very large counts (e.g., 107, 140, 148), supporting rigorous evaluation in both sparse and dense settings.
- **Diverse backgrounds:** Prompts encompass a wide array of real-world contexts, such as *in a kitchen cabinet, on a picnic table, on a pantry shelf, on a couch armrest, in the sky, in the water, over a valley, on a refrigerator, on a lunch tray*, etc.
- **Composite categories:** Multi-category prompts combine classes (e.g., cats and dogs, balloons and pineapples, bears and mice, cats and suitcases, candles and donuts, cars and helicopters), enabling compositional reasoning beyond

single-object scenes.

A brief statistics of our benchmark is shown in Fig. 9.

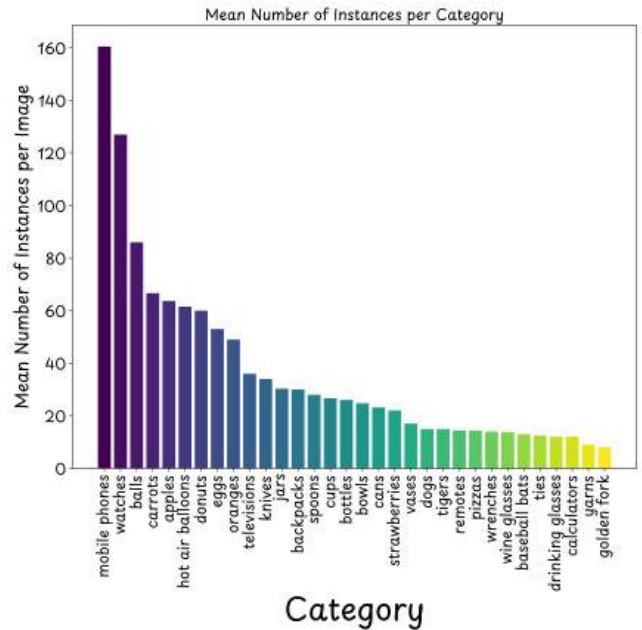


Figure 9: Statistics (instance per image vs category) for the COUNTLOOP-S benchmark.

Details on Human Evaluation Setup: We designed our human evaluation survey using Google Forms. Raters were asked to evaluate three images per set in terms of prompt alignment, aesthetic quality, and overall preference. A total of 15 image sets were selected across all four benchmarks, covering diverse prompts, object categories, and scene complexities to ensure representative assessment. Participants (N=30) had an average age of 31 (range 22–45), and came from professional backgrounds in graphic design (20), AI art and research (10). Approximately 10 participants had prior experience or domain expertise in tasks requiring precise object counting (e.g., data annotation, inventory management, or computer vision evaluation).

Additional Ablation Details

We perform additional ablations on our model design on COCO-Count benchmark.

Runtime Analysis: We evaluated the runtime required by COUNTLOOP and one of the best-performing agentic systems, SLD (Wu, Lian et al. 2023), to achieve the exact target object count on prompts containing 10, 50, and 100 instances. For each configuration, we averaged over 10 independent runs and report both the mean and standard deviation. In all cases, COUNTLOOP converges significantly faster than SLD, with speedups of approximately 1.2× at low counts and up to 1.4× at high counts. In addition to this, SLD performs poorly in single/multi-category multi-instance scenarios as shown in Table 1 (main paper), demonstrating the superiority of our approach both in terms of efficiency and image quality.

Image Evaluation Study

Welcome to this image evaluation study! Your task is to review sets of images and provide feedback on two key aspects:

- Prompt Alignment:** How well does the image match the description in the prompt?
- Aesthetic Quality:** How visually appealing is the image?

For each prompt, you will see three images labeled A, B, and C. Please answer the questions for each set and indicate your overall preference.

Prompt Alignment is a measure of how accurately a generated image reflects the description in the prompt. It ensures the correct number of objects (e.g., 5 apples), the accurate depiction of the setting and arrangement (e.g., apples on a wooden table), and the inclusion of any additional details. When rating, evaluate both objective elements, like object count, and subjective aspects, like setting, on a scale from 1 (Poor alignment) to 5 (Excellent alignment).

Aesthetic Quality assesses the visual appeal of the image, considering factors such as composition, clarity, color balance, and overall harmony. A high-quality image is clear, well-composed, and pleasing to the eye, with vibrant or appropriate colors and no noticeable distortions. When rating, consider how the image's visual elements come together on a scale from 1 (Poor quality, e.g., blurry or unbalanced) to 5 (Excellent quality, e.g., sharp and visually striking).

Your feedback is anonymous and greatly appreciated.

* Indicates required question


Do you agree to proceed with the survey? *

☐ Yes

☐ No

Section 1

A photo of 2 dogs and 8 balloons in a room



Prompt Alignment: Rate how well each image aligns with the prompt (1 = Poor alignment, 5 = Excellent alignment).

	1	2	3	4	5
Image A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Image B	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Image C	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Aesthetic Quality: Rate the aesthetic quality of each image (1 = poor, 5 = excellent).

	1	2	3	4	5
Image A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Image B	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Image C	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Overall Preference: Which image do you prefer overall, considering both prompt alignment and aesthetic quality?

☐ Image A

☐ Image B

☐ Image C

Figure 10: Screenshot of the Google Form used for human evaluation.

Table 4: Runtime comparison for achieving correct count (mean \pm std, in seconds)

Model	10 instances	50 instances	100 instances
SLD (Wu, Lian et al. 2023)	35.2 \pm 1.8	110.5 \pm 3.2	165.8 \pm 4.5
COUNTLOOP (Ours)	28.4 \pm 1.2	75.3 \pm 2.7	120.1 \pm 3.9

Impact of Iterative Refinement Rounds: We investigate the impact of varying the number of iterations on both counting accuracy and aesthetic quality using the COCO-Count split. Experiments range from a single iteration (baseline) to two iterations. We show in Table 5 that while a single iteration achieves reasonable results, additional iterations significantly improve both counting accuracy and aesthetic quality.

Table 5: Ablation on the number of iterations.

Iterations	Counting (%) \uparrow		Spatial \uparrow
	F1	Acc.	
1	89.72	85.44	0.79
2	98.47	93.33	0.93

Impact on Object-Aware Attention Expansion: In order to isolate the contribution of our object-aware self-attention (attention expansion) mechanism (see Eq. 5 in the main text), we conducted an ablation study on the COCO-Count benchmark. In this variant, we remove the attention-expansion module: after generating all instance-wise query representations Z_i^q with the IP-Adapter (Eq. 4), we skip the joint self-attention among these queries (Eq. 5) and directly decode each object independently, followed by background inpainting. As Table 6 shows, omitting attention expansion causes a noticeable

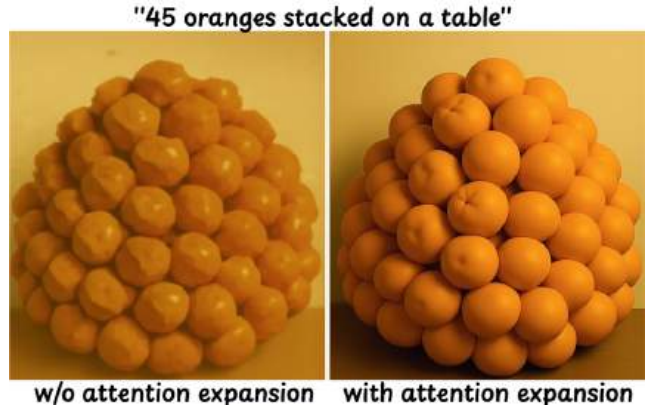


Figure 11: Illustrating the effect of attention expansion. The quality of generated objects degrades when images are allowed to generate directly using cumulative features alone.

drop in spatial coherence – measured by the CLIP-FlanT5 aesthetic score – while the object counting F1 and accuracy remain effectively unchanged. This is visually illustrated in Fig. 11, where without attention expansion leads to degradation in semantic due to leakage. This confirms that attention expansion primarily enhances inter-instance semantic consistency without affecting count control.

Table 6: Effect of object-aware attention expansion on COCO-Count. “Spatial” is the CLIP-FlanT5 aesthetic score.

Model Variant	Counting F1 (%)	Counting Acc. (%)	Spatial
Full COUNTLOOP	98.47	93.33	0.93
w/o Attention Expansion	85.39	81.07	0.82

Performance with different LLM backbones: In this section, we compare the performance of different open-sourced large language models (LLMs) on the COCO-Count benchmark. Table 7 presents the results for Qwen3 (Yang et al. 2025), Pixtral (Agrawal et al. 2024), and LLaVA (Liu et al. 2023), with Qwen outperforming the others in both metrics.

Table 7: Performance comparison of different LLMs

Model	F1 Score \uparrow	Spatial \uparrow
LLaVA	90.87	0.88
Pixtral	92.13	0.91
Qwen	98.47	0.93

Count Accuracy under different numbers of instances:: We plot counting accuracy vs number of instances in Fig. 12. As seen, COUNTLOOP delivers consistently better performance than most benchmarks, even with high instances (100)

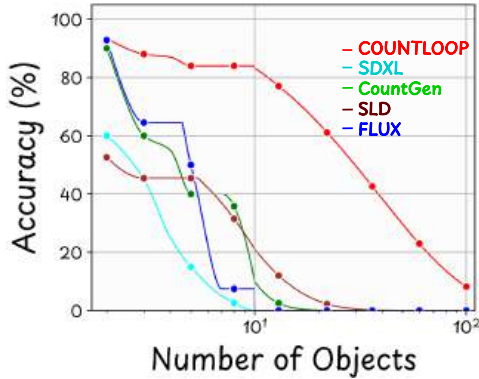


Figure 12: Count accuracy vs number of objects per image for COUNTLOOP, SDXL, Make-It-Count, SLD, and FLUX

Performance with Different Diffusion Backbones:

To assess the generality of COUNTLOOP across diffusion backbones, we replaced the default SDXL model with two additional Stable Diffusion checkpoints: *Stable Diffusion v1.5* (sd-v1-5.ckpt) and *Stable Diffusion 3.5* (sd3.5-base). We kept all other components (planning graph, cumulative attention, IP-Adapter, critic loop) and hyperparameters identical. Table 8 reports counting F1, exact-match accuracy, and spatial scores on the COCO-Count benchmark. While all backbones benefit substantially from COUNTLOOP’s structured refinement, we observe that higher-capacity models yield marginally better spatial coherence, with SDXL at the top. Importantly, counting performance remains robust ($F1 \geq 97.2\%$) across backbones, demonstrating that COUNTLOOP’s instance-control mechanism is largely model-agnostic.

Additional Qualitative Results

Here we provide some additional results of the LLM and the Image generation pipeline along with an application of COUNTLOOP.

Table 8: COUNTLOOP’s performance with different Stable Diffusion backbones on COCO-Count.

Backbone	Counting F1 (%)	Counting Acc. (%)	Spatial
SD v1.5 (sd-v1-5.ckpt)	97.21	91.05	0.88
SD 3.5 (sd3.5-base)	97.95	92.10	0.90
SDXL (sdxl-base-1.0)	98.47	93.33	0.93

LLM Prompt Template: We have provided the prompt templates for LLM instructions used in Sec 3. More specifically, we have provided the expanded prompt instructions used for (a) In-Context Learning Prompt for Layout Generation and (b) Prompt for Design Critic LLM, respectively.

Qualitative Comparison Analysis: In addition to the qualitative results presented in the main paper, we have also provided a qualitative comparison (Fig. 13) and a generation gallery (Fig. 14). The visual results provide compelling evidence of COUNTLOOP’s effectiveness in high-instance generation against SoTA models, under both single and multiple category scenarios.

Style-Aligned Image Generation

A pretrained diffusion U-Net model fine-tuned with LoRA (Low-Rank Adaptation) can produce vastly different visual styles from the same base concept. For example, the “13 cats” in Fig. 15 maintain the subject’s constant while each panel applies a distinct style (photorealistic, semi-realistic 3D, anime, oil painting, sci-fi concept art, and storybook illustration), altering the lighting and rendering approach without altering the core content. Under the hood, LoRA fine-tuning freezes the original diffusion model’s weights and inserts a small set of trainable low-rank matrices into the network. These low-rank weight updates capture the new style’s visual patterns (e.g., realistic fur vs. flat cartoon shading) without having to modify all of the model’s parameters. This parameter-efficient approach enables fast, memory-light adaptation to each style, essentially a learned style transfer inside the diffusion process, while preserving the model’s base knowledge (how to depict cats). Crucially, only a few additional parameters (on the order of megabytes) are required for each style, allowing each stylistic variation to be achieved without retraining or duplicating the entire multi-gigabyte models.

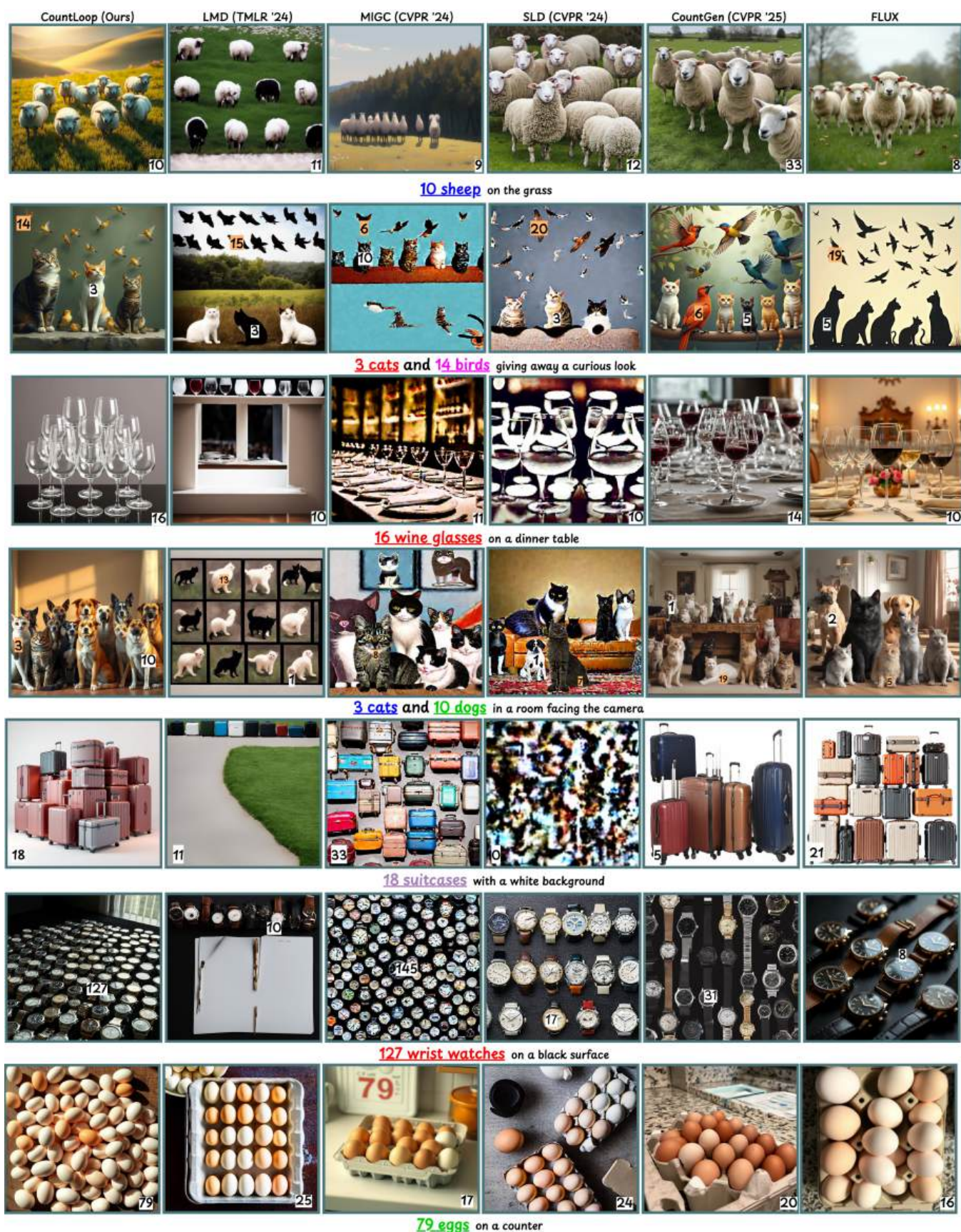


Figure 13: Comparison with SoTA



9 pineapples and 2 lions



3 birds and 4 dogs



4 cups, 4 saucers, and 6 sugar cubes on the table



34 jars in a kitchen cabinet



5 cans and 17 donuts in a kitchen cabinet



5 cupcakes and 7 balloons on a table



12 apples in X shape on a table



42 bowls in a kitchen shelf



107 apples illuminated by 12 candles



10 carrots and 10 apples on a table



8 cats and 11 bananas



21 pizzas on a serving plate



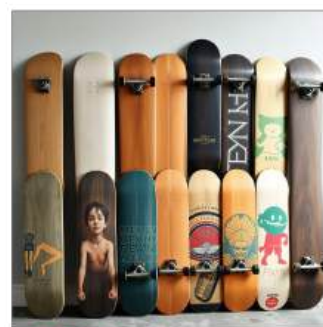
43 cans on a pantry shelf



30 backpacks on a dark surface



29 bottles in the fridge



15 skateboards of various shapes leaning against a wall

Figure 14: Visuals from our COUNTLOOP-M & COUNTLOOP-S benchmarks using COUNTLOOP .

In-Context Learning Prompt for Layout Generation

SYSTEM: You are a Layout Designer AI specialized in converting text prompts into detailed spatial layouts for image generation.

CRITICAL INSTRUCTIONS:

1. Assign natural, non-grid positions
2. Include depth information (d)
3. Calculate realistic spatial relationships
4. Maintain proportional object sizes
5. Output ONLY valid JSON

JSON SCHEMA:

```
{
  "objects": [
    {
      "id": "unique identifier",
      "pos": [x, y],
      "d": depth,
      "size": [w, h],
      "color": "primary color",
      "attributes": ["list", "of", "attributes"]
    }
  ],
  "relations": [
    {
      "from": "source id",
      "to": "target id",
      "relation": "spatial relation",
      "dist": pixel_distance,
      "angle": degrees
    }
  ],
  "context": "background description"
}
```

EXAMPLE 1:

PROMPT: "2 cats and 1 bird in sky"

```
{
  "objects": [
    {"id": "cat 1", "pos": [0.3, 0.6], "d": 0.4, "size": [0.2, 0.25]},
    {"id": "cat 2", "pos": [0.6, 0.65], "d": 0.4, "size": [0.22, 0.27]},
    {"id": "bird 1", "pos": [0.5, 0.3], "d": 0.2, "size": [0.15, 0.1]}
  ],
  "relations": [
    {"from": "cat 1", "to": "bird 1", "relation": "below", "dist": 120, "angle": 90},
    {"from": "cat 2", "to": "bird 1", "relation": "below", "dist": 100, "angle": 85}
  ],
  "context": "outdoor, grassy field"
}
```

EXAMPLE 2:

PROMPT: "15 identical watches on stand"

```
{
  "objects": [
    {"id": "watch 1", "pos": [0.15, 0.4], "d": 0.7, "size": [0.06, 0.06]},
    {"id": "watch 2", "pos": [0.22, 0.42], "d": 0.7, "size": [0.06, 0.06]},
    // ... additional watches
    {"id": "watch 15", "pos": [0.85, 0.45], "d": 0.7, "size": [0.06, 0.06]}
  ],
  "relations": [
    {"from": "watch 1", "to": "watch 2", "relation": "right of", "dist": 45, "angle": 10},
    // ... additional relations
  ],
  "context": "wooden display stand"
}
```

EXAMPLE 3 (High-Count):

PROMPT: "107 identical balloons"

```
{
  "objects": [
    {"id": "balloon 1", "pos": [0.12, 0.25], "d": 0.3, "size": [0.04, 0.04]},
    {"id": "balloon 2", "pos": [0.15, 0.28], "d": 0.35, "size": [0.04, 0.04]},
    // ... 103 additional balloons
    {"id": "balloon 107", "pos": [0.88, 0.45], "d": 0.25, "size": [0.04, 0.04]}
  ],
  "relations": [
    {"from": "balloon 1", "to": "balloon 2", "relation": "right of", "dist": 30, "angle": 15},
    // ... key spatial relations
  ],
  "context": "clear blue sky"
}
```

CURRENT PROMPT: "{PROMPT}"

OUTPUT:

Prompt for Design Critic LLM

SYSTEM: You are a Designer Critic AI that evaluates generated images against prompts and provides structured feedback for refinement. Analyze object count, spatial arrangement, and aesthetics, then output actionable feedback in JSON.

INSTRUCTIONS:

- Analyze ONLY the provided image, prompt, and score
- Identify SPECIFIC issues with object IDs/positions
- Provide CONCRETE fixes (not vague suggestions)
- Prioritize count accuracy issues
- Output ONLY valid JSON

JSON SCHEMA:

```
{
  "evaluation": {
    "count_accuracy": {"detected": int, "target": int},
    "spatial_quality": float,
    "decision": {
      "continue_refinement": boolean,
      "reason": "justification"
    }
  }
}
```

EXAMPLE (High-Count):

PROMPT: "15 watches on display stand"

GENERATED: 12 watches in grid pattern

```
{
  "evaluation": {
    "count_accuracy": {"detected": 12, "target": 15},
    "spatial_quality": 0.6
  },
  "issues": [
    {
      "type": "count",
      "severity": "critical",
      "description": "3 watches missing",
      "suggested_fix": "Add watches 13-15 at [0.72,0.41], [0.79,0.39], [0.86,0.43]"
    }
  ],

```

```
{
  "type": "spatial",
  "severity": "major",
  "description": "Artificial grid pattern",
  "suggested_fix": "Vary spacing (42-48px) and angles (-3 to +10^{\circ})"
},
{
  "decision": {
    "continue_refinement": true,
    "reason": "Count error requires refinement"
  }
}
```

CURRENT PROMPT: "{PROMPT}"

CURRENT LAYOUT: "{LAYOUT}"

OUTPUT:



13 cats on a wooden shelf, **photorealistic**, **natural lighting**, **detailed fur**, **indoor setting**



13 cats in a clean studio, **semi-realistic 3D render**, **soft lighting**, **stylized but natural**



13 cartoon cats with big eyes under a starry sky, **anime-style**, **flat shading**, **outlined**



13 cats portrait in **classical oil painting style**, **warm tones**, **visible brushstrokes**



13 futuristic cats with robot armor, **sci-fi concept art**, **cyberpunk lighting**, **metallic details**



13 whimsical group of cats on bookshelves, **soft colors**, **storybook illustration style**

Figure 15: COUNTLOOP's style control capability

References

- Agrawal, P.; Antoniak, S.; Hanna, E. B.; Bout, B.; Chaplot, D.; Chudnovsky, J.; Costa, D.; De Monicault, B.; Garg, S.; Gervet, T.; et al. 2024. Pixtral 12B. *arXiv preprint arXiv:2410.07073*.
- Alimohammadi, A.; Nag, S.; Taghanaki, S. A.; Tagliasacchi, A.; Hamarneh, G.; and Amiri, A. M. 2024. SMITE: Segment Me In TimE. *arXiv preprint arXiv:2410.18538*.
- Amazon Ads. 2023. Revolutionizing creative production with AI image generation. <https://advertising.amazon.com/blog/ai-image-generation>. Accessed: 2025-05-14.
- Barron, J. 2025. Tweet about bias in GPT-4o image generation. https://x.com/jon_barron/status/1915828262326178145. "....4o has a bias towards putting things in the upper left of the image....." (Accessed: 2025-07-31).
- Betker, J.; Goh, G.; Jing, L.; Brooks, T.; Wang, J.; Li, L.; Ouyang, L.; Zhuang, J.; Lee, J.; Guo, Y.; et al. 2023. Improving image generation with better captions. *Computer Science*, 2(3): 8. <https://cdn.openai.com/papers/dall-e-3.pdf>.
- Binyamin, L.; Tewel, Y.; et al. 2024. Make It Count: Text-to-Image Generation with an Accurate Number of Objects. *ArXiv:2406.10210*.
- Black-Forest-Labs. 2024. FLUX. <https://github.com/black-forest-labs/flux>.
- Chefer, H.; Alaluf, Y.; Vinker, Y.; and Cohen-Or, D. 2023a. Attend-and-Excite: Probabilistic Attention Guidance for Diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3321–3331. New York, NY, USA: IEEE.
- Chefer, H.; Alaluf, Y.; Vinker, Y.; Wolf, L.; and Cohen-Or, D. 2023b. Attend-and-Excite: Attention-Based Semantic Guidance for Text-to-Image Diffusion Models. *ACM Trans. Graph.*, 42(4).
- Chen, D.; Chen, R.; Pu, S.; Liu, Z.; Wu, Y.; Chen, C.; Liu, B.; Huang, Y.; Wan, Y.; Zhou, P.; et al. 2024. Interleaved scene graphs for interleaved text-and-image generation assessment. *arXiv preprint arXiv:2411.17188*.
- Cheng, J.; Yin, B.; Cai, K.; Huang, M.; Li, H.; He, Y.; Lu, X.; Li, Y.; Li, Y.; Cheng, Y.; et al. 2024. Theatergen: Character management with llm for consistent multi-turn image generation. *arXiv preprint arXiv:2404.18919*.
- Dahary, O.; Cohen, Y.; Patashnik, O.; Aberman, K.; and Cohen-Or, D. 2025. Be Decisive: Noise-Induced Layouts for Multi-Subject Generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, 1–12.
- Dahary, O.; Patashnik, O.; Aberman, K.; and Cohen-Or, D. 2024a. Be Yourself: Bounded Attention for Multi-Subject Text-to-Image Generation. *ArXiv:2403.16990*.
- Dahary, O.; Patashnik, O.; Aberman, K.; and Cohen-Or, D. 2024b. Be Yourself: Bounded Attention for Multi-Subject Text-to-Image Generation. In *Proceedings of the European Conference on Computer Vision*, 432–448. Springer, Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-031-72629-3.
- Feng, R.; Miao, Y.; and Zheng, J. 2024. A YOLO-Based Intelligent Detection Algorithm for Risk Assessment of Construction Sites. *Journal of Intelligent Construction*, 2(4): 1–18.
- Feng, W.; Zhu, W.; Fu, T.-J.; et al. 2023. LayoutGPT: Compositional Visual Planning and Generation with Large Language Models. *ArXiv:2305.15393*.
- Huang, K.; Sun, K.; Xie, E.; Li, Z.; and Liu, X. 2023. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36: 78723–78747.
- Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, -(-): –.
- Johnson, J.; Gupta, A.; and Fei-Fei, L. 2018. Image Generation from Scene Graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1219–1228. Salt Lake City, UT, USA: IEEE.
- Kang, W.; Galim, K.; and Koo, H. I. 2023. Counting Guidance for High-Fidelity Text-to-Image Synthesis. *ArXiv:2309.04666*.
- Li, B.; Lin, Z.; Pathak, D.; Li, J.; Fei, Y.; Wu, K.; Xia, X.; Zhang, P.; Neubig, G.; and Ramanan, D. 2024. Evaluating and Improving Compositional Text-to-Visual Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5290–5301. Seattle, WA, USA: IEEE.
- Li, Y.; Liu, H.; Yang, J.; et al. 2023. GLIGEN: Open-set Grounded Text-to-Image Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22511–22521. Los Alamitos, CA, USA: IEEE Computer Society.
- Lian, L.; Li, B.; Yala, A.; and Darrell, T. 2023. LLM-grounded Diffusion: Enhancing Prompt Understanding of Text-to-Image Diffusion Models with Large Language Models. *Trans. Mach. Learn. Res.*, 2024.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, 740–755. Zurich, Switzerland: Springer, Cham.
- Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36: 34892–34916.
- Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Jiang, Q.; Li, C.; Yang, J.; Su, H.; et al. 2024. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. In *Proceedings of the European Conference on Computer Vision*, 38–55. Milan, Italy: Springer.
- Mondal, A.; Nag, S.; Zhu, X.; and Dutta, A. 2025. Omnicount: Multi-label object counting with semantic-geometric priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 19537–19545. -: AAAI.

- Podell, D.; English, Z.; Lacey, K.; Blattmann, A.; Dockhorn, T.; Müller, J.; Penna, J.; and Rombach, R. 2024. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*, -. -: OpenReview.net.
- Podell, D.; Liu, H.; et al. 2023. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. ArXiv:2307.01952.
- Ramachandran, R.; Garjani, A.; Bachmann, R.; Atanov, A.; Kar, O. F.; and Zamir, A. 2025. How Well Does GPT-4o Understand Vision? Evaluating Multimodal Foundation Models on Standard Computer Vision Tasks. *arXiv preprint arXiv:2507.01955*.
- Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10674–10685. Los Alamitos, CA, USA: IEEE Computer Society.
- Saharia, C.; Chan, W.; Saxena, S.; Lit, L.; Whang, J.; Denton, E.; Ghasemipour, S. K. S.; Ayan, B. K.; Mahdavi, S. S.; Gontijo-Lopes, R.; Salimans, T.; Ho, J.; Fleet, D. J.; and Norouzi, M. 2022. Photorealistic text-to-image diffusion models with deep language understanding. In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713871088.
- Stability-AI. 2025. sd3.5. <https://github.com/Stability-AI/sd3.5>.
- Team, A. C. 2023. Creative Trends 2023: Insights for Designers. *Adobe Blog*.
- Wang, Z.; Li, A.; Li, Z.; and Liu, X. 2024. Genartist: Multimodal llm as an agent for unified image generation and editing. *Advances in Neural Information Processing Systems*, 37: 128374–128395.
- Wu, H.; Zhang, Z.; Zhang, W.; Chen, C.; Liao, L.; Li, C.; Gao, Y.; Wang, A.; Zhang, E.; Sun, W.; Yan, Q.; Min, X.; Zhai, G.; and Lin, W. 2024. Q-ALIGN: teaching LMMs for visual scoring via discrete text-defined levels. In *Proceedings of the 41st International Conference on Machine Learning, ICML'24*. Vienna, Austria: JMLR.org.
- Wu, J. Z.; Ge, Y.; Wang, X.; Lei, S. W.; Gu, Y.; Shi, Y.; Hsu, W.; Shan, Y.; Qie, X.; and Shou, M. Z. 2023. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 7623–7633.
- Wu, T.-H.; Lian, L.; et al. 2023. Self-Correcting LLM-Controlled Diffusion Models. ArXiv:2311.16090.
- Xiong, J.; Liu, G.; Huang, L.; Wu, C.; Wu, T.; Mu, Y.; Yao, Y.; Shen, H.; Wan, Z.; Huang, J.; et al. 2024. Autoregressive models in vision: A survey. *arXiv preprint arXiv:2411.05902*.
- Yan, Z.; Ye, J.; Li, W.; Huang, Z.; Yuan, S.; He, X.; Lin, K.; He, J.; He, C.; and Yuan, L. 2025. GPT-ImgEval: A Comprehensive Benchmark for Diagnosing GPT-4o in Image Generation. *arXiv preprint arXiv:2504.02782*, -(-): -.
- Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, -(-): -.
- Yang, L.; Yu, Z.; Meng, C.; Xu, M.; Ermon, S.; and Cui, B. 2024. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. In *Forty-first International Conference on Machine Learning*.
- Yap, J. A. 2024. 15 Extinct Animals Imagined by Midjourney (AI Art). <https://goldpenguin.org/blog/extinct-animals-imagined-by-midjourney-ai-art/>. Updated June 14, 2024. Available at: <https://goldpenguin.org/blog/extinct-animals-imagined-by-midjourney-ai-art/>.
- Ye, H.; Zhang, J.; Liu, S.; Han, X.; and Yang, W. 2023. IP-Adapter: Text Compatible Image Prompt Adapter for Text-to-Image Diffusion Models. *arXiv preprint arXiv:2308.06721*, -(-): -.
- Yuksekgonul, M.; Bianchi, F.; Boen, J.; Liu, S.; Lu, P.; Huang, Z.; Guestrin, C.; and Zou, J. 2025. Optimizing generative AI by backpropagating language model feedback. *Nature*, 639(8055): 609–616.
- Zhang, W.; et al. 2024. SceneLayoutNet: Generating Images from Scene Layouts Using Transformers. In *Advances in Neural Information Processing Systems*, -. -: -.
- Zhou, D.; Li, Y.; Ma, F.; Zhang, X.; and Yang, Y. 2024. MIGC: Multi-Instance Generation Controller for Text-to-Image Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6818–6828. -: IEEE.