Think, Plan, Paint: Layout-Aware Reasoning for Controllable Image Generation in Unified Models

Junhao Liu[1,*] **Jian-Wei Zhang**[2] **Tao Huang**[2]
Miles Yang[2] **Zhao Zhong**[2,†] **Liefeng Bo**[2]
[1]Key Lab of HCST (PKU), MOE; SCS, Peking University, China
[2]Tencent Hunyuan, China
liujunhao@pku.edu.cn

Abstract

Unified Multimodal Large Language Models (MLLMs) offer a promising paradigm for unifying visual understanding and generation, yet they still struggle to follow complex spatial instructions and logical constraints in controllable image generation. To address this gap, we present ATLAS, a unified framework that equips MLLMs with a human-like "Think, Plan, and Paint" paradigm. We adopt layout as the shared representation that connects the three stages, enabling the model to reason about spatial requirements, plan explicit object arrangements, and render the final image. We further improve plan-to-image fidelity with reinforcement-learning-based layout alignment. We instantiate ATLAS at 7B and 80B scales, achieving state-of-the-art performance among MLLMs on image generation benchmarks and an average 65.31% improvement over existing layout-based unified MLLMs. On spatially related tasks, ATLAS obtains an average 23.06% gain over the base models. Through the same layout interface, ATLAS also supports instruction-guided editing and multimodal grounding. We further introduce ATLAS-Reasoning, a benchmark for evaluating generation under complex spatial instructions.

1 Introduction

In image generation, a promising direction is to introduce *Layout* as an intermediate representation to bridge the gap between abstract textual instructions and pixel-level arrangements, enabling more controllable image generation [17, 53, 13]. Meanwhile, unified Multimodal Large Language Models (MLLMs) have emerged as a promising paradigm for image understanding and generation [8, 11, 6]. He et al. [18] have taken an initial step toward integrating layout planning and image generation into unified MLLMs. However, this line of work faces two critical limitations in practice. First, layout planning is often performed as a direct prediction from the prompt, without an explicit reasoning process to decompose objects, counts, and spatial relations, which can lead to incomplete or inconsistent layouts under complex instructions. Second, layout is still treated mainly as an auxiliary control signal for generation, rather than a shared spatial representation that can consistently support planning, generation, editing, and understanding.

To address these challenges, we argue that the potential of unified MLLMs lies in treating layout as a cross-modal bridge rather than a mere control signal, which enables the model to leverage the reasoning strengths of the text modality to structure the visual modality. In this view, the explicit layout representation serves as a transparent window into the generation process, converting

[*]Work done during internship at Tencent Hunyuan.
[†]Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

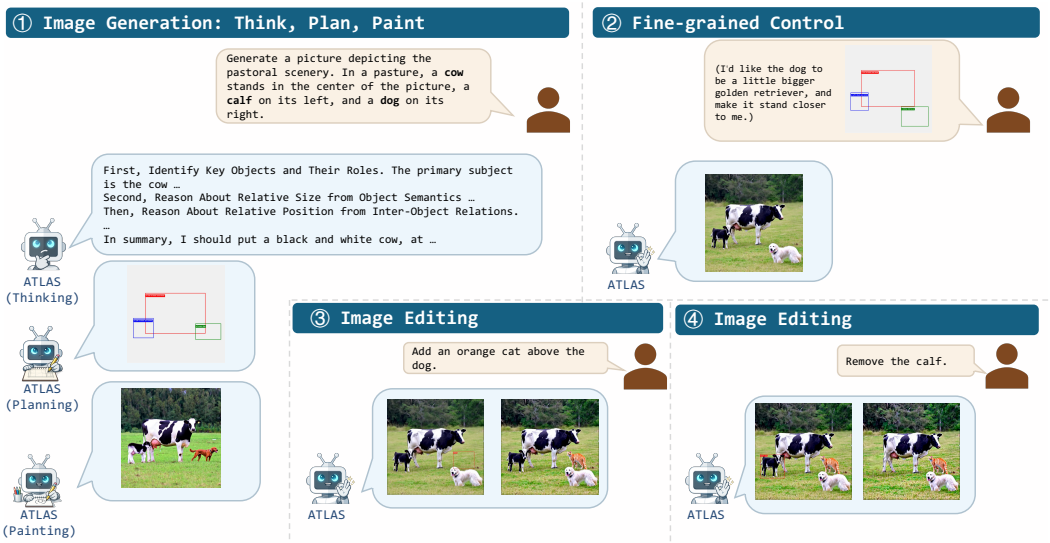


Figure 1: Our ATLAS framework generates images following a human-like *Think, Plan, and Paint* paradigm, and supports interleaved image generating and editing. Additionally, the Think and Plan phases provide the user with fine-grained control over the generation process.

the "black-box" workflow into an interpretable one, and functions as an interactive interface that empowers users to exert fine-grained control over the final synthesis.

Motivated by this view, we propose **ATLAS** (**A**ligned **T**hinking **L**ayout-**A**ware image **S**ynthesis), a framework that enables unified MLLMs to generate images through a human-like ***"Think, Plan, Paint"*** paradigm as shown in Figure 1.

Specifically, ATLAS initiates a ***Think*** process to analyze object semantics and spatial constraints before executing the ***Plan*** phase. To operationalize the cross-modal bridge role of layout, we introduce Shared Positional Tokens as a unified spatial vocabulary across understanding, planning, generation, and editing, and further apply Reinforcement Learning (RL)-based layout alignment to the ***Paint*** stage so generated objects better align with the planned layout. These explicit intermediate states make generation more interpretable and editable, while the Transfusion-style ***Paint*** stage turns the planned layout into high-fidelity images.

We implement ATLAS on two unified auto-regressive Transfusion-style backbones [11, 6] and verify the effectiveness and scalability of our approach by instantiating ATLAS at two scales: **7B** and **80B**. Our experiments demonstrate that ATLAS outperforms existing layout-based unified MLLMs. In particular, our **80B** model achieves comparable generation performance to leading closed-source large models.

In summary, our contributions are:

- We propose **ATLAS**, a unified MLLM framework with the ***"Think, Plan, Paint"*** paradigm. We leverage the reasoning strengths of the text modality to structure the visual modality, enhancing the capability to handle complex spatial instructions while enabling precise, fine-grained user editing.
- We empower the ***Think*** and ***Plan*** phases with precise spatial grounding to ensure logical validity. By establishing a unified spatial vocabulary via Shared Positional Tokens and further aligning the ***Paint*** stage with RL-based layout alignment, we improve the consistency between coherent plans and generated objects.
- We instantiate ATLAS at **7B** and **80B** scales. ATLAS outperforms existing layout-based unified MLLMs, and the **80B** model achieves comparable generation performance to leading closed-source large models.
- We introduce **ATLAS-Reasoning**, a new benchmark designed to rigorously evaluate reasoning-aware generation and layout planning under complex spatial and logical instructions.

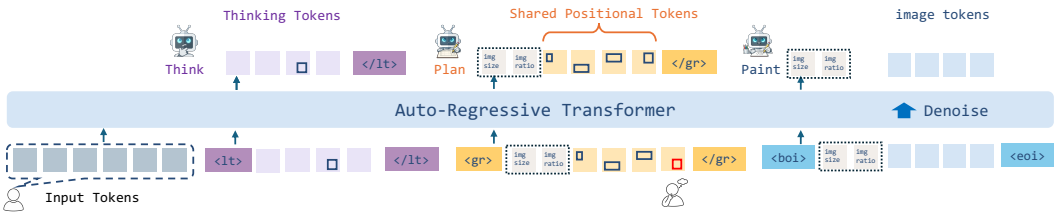


Figure 2: Overview of the ATLAS framework. Given a text prompt, ATLAS *Thinks* through object semantics and spatial relations, *Plans* a layout with Shared Positional Tokens that align text and vision and allow precise user edits, then *Paints* a final image that faithfully follows the planned composition. Shared Positional Tokens represent object layouts in both *Think* and *Plan* phases.

2 Background and Related Work

2.1 Unified Multimodal Large Language Models

Unified Multimodal Large Language Models (MLLMs) are a mainstream paradigm for multimodal tasks. While early models[28, 2, 35, 1] focused primarily on understanding, recent works[42, 36, 8, 15, 27] have unified input and output modalities, enabling arbitrary-to-arbitrary generation. A critical architectural evolution in this domain is the transition from discrete vector quantization to continuous representations. While discrete models often struggle with visual fidelity, recent hybrid architectures integrate AR-transformers with diffusion models [55, 11, 6, 45].

2.2 Controllable Image Generation and Manipulation

Understanding and controlling model behavior have long been central concerns in machine learning [30, 31, 33, 32, 48, 49, 40, 41]. In visual synthesis, this pursuit has evolved beyond simple text-to-image mapping. Early approaches focused on image manipulation, using text instructions to edit existing images [5, 19, 14], or employing dense conditions (e.g., depth maps, canny edges) for structural control [52, 37]. Among these interfaces, **Layouts** (e.g., bounding boxes) stand out as the most intuitive and explicit representation for defining spatial composition. Current layout-driven approaches generally fall into three paradigms:

Adapter-based methods [52, 26, 7, 44] inject spatial conditions into frozen diffusion models. While effective, these methods are inherently *passive*, requiring pre-defined layouts and lacking the intrinsic capability to "plan" from text alone.

Pipeline approaches [9, 13, 53, 54] address this by using external LLMs to infer layouts for downstream generators. However, this creates a *disjointed* system where planning is isolated from visual perception, leading to error propagation.

Unified approaches like PlanGen [18] step forward by integrating planning and generation within a single model.

Yet, as discussed below, they often treat layout generation as a direct translation task, overlooking the necessity of explicit reasoning, thus limiting their ability mainly to layout-guided generation, while struggling with layout planning.

3 The ATLAS Framework

In this section, we introduce ATLAS in detail. Figure 2 and Figure 3 illustrate the overall architecture and workflow of ATLAS.

3.1 Unified Architecture

3.1.1 Auto-regressive Transformer Backbone

We instantiate ATLAS on two auto-regressive MLLM backbones, **BAGEL** [11] and **Hunyuan Image 3.0** [6]. Both models represent text and image signals in a shared sequence, where text is encoded

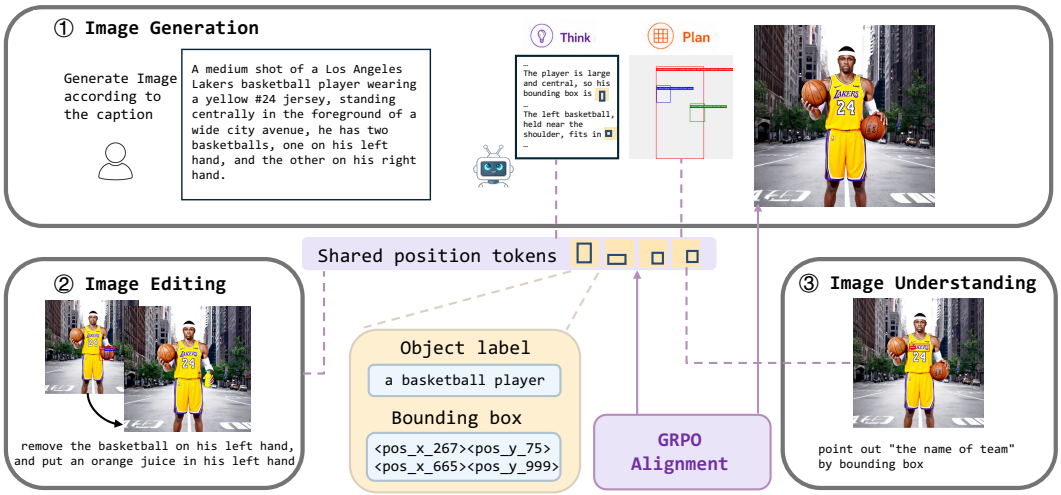


Figure 3: ATLAS represents each grounded object with a semantic label followed by four dedicated coordinate tokens from the shared vocabulary V_{pos}. The same label-box format serves as a unified spatial interface across image generation, image editing, and multimodal understanding. We further apply reinforcement learning to align the planned layout with the generated image.

as discrete embeddings and images are processed as continuous latent variables. Using these two backbones allows us to evaluate ATLAS across different architectures and parameter scales.

3.1.2 Shared Positional Tokens

As illustrated in Figure 3, we introduce **Shared Positional Tokens** as a unified spatial interface for image generation, image editing, and multimodal understanding. Each grounded object is represented as a label-box pair (c_i, b_i), where c_i is the object label and $b_i = (x_i^1, y_i^1, x_i^2, y_i^2)$ denotes the top-left and bottom-right corners of its bounding box. We use a dedicated coordinate-token vocabulary V_{pos} for spatial positions. Specifically, each normalized coordinate $u \in [0, 1]$ is assigned to one of 1000 position indices by $q(u) = \min(\lfloor 1000u \rfloor, 999)$, and we define $V_{pos} = V_x \cup V_y$, where $V_x = \{\texttt{<pos_x_0>}, \ldots, \texttt{<pos_x_999>}\}$ and $V_y = \{\texttt{<pos_y_0>}, \ldots, \texttt{<pos_y_999>}\}$. Let $\tau_x(k)$ and $\tau_y(k)$ denote the corresponding x- and y-axis tokens in V_{pos}. A box is then expressed as $\tau_x(q(x_i^1)), \tau_y(q(y_i^1)), \tau_x(q(x_i^2)), \tau_y(q(y_i^2))$, such as <pos_x_267><pos_y_75><pos_x_665><pos_y_999> in Figure 3.

3.2 The "Think, Plan, Paint" Paradigm

To fully leverage unified MLLM's capabilities, we structure the generation process into three explicit phases, transforming layout generation from an intuitive mapping into a reasoned deduction.

3.2.1 Think: Layout-Aware Chain-of-Thought

Before committing to any spatial arrangement, ATLAS initiates a **Think** phase. Instead of blindly mapping text to coordinates, the model decomposes complex scenes into individual entities, deduces implicit spatial constraints with the capabilities of textual reasoning, and also takes visual priors into account by leveraging the Shared Positional Tokens. This Layout-Aware Chain-of-Thought (CoT) serves as a cognitive foundation, enhancing the model's ability to plan coherent layouts.

3.2.2 Plan: Grounded Layout Generation

Conditioned on the reasoning chain R, the model enters the **Plan** phase and generates a layout sequence $L = \{(c_i, b_i)\}_{i=1}^{n}$, where each pair follows the label-box token format defined in Section 3.1.2. Rather than directly predicting coordinates from the prompt alone, this phase uses the preceding reasoning to determine which objects should be grounded and how their boxes should satisfy the

inferred spatial relations, counts, and attributes. The resulting layout L serves as an executable scene plan that can be inspected or edited by users and directly consumed by the subsequent *Paint* phase.

3.2.3 Paint: Conditioned Image Synthesis

In the **Paint** phase, the model synthesizes the image I. The backbone denoises the random latent noise conditioned on the entire preceding sequence: the prompt T, the reasoning R, and the planned layout L. By attending to the precise layout tokens generated in the Plan phase, the model "paints" pixels that faithfully respect the specified spatial structure.

3.3 Data Construction and Training Objectives

To equip the model with this unified paradigm, we construct a specialized instruction-tuning dataset and employ a joint optimization strategy.

3.3.1 Data Construction

To empower ATLAS with the "Think, Plan, Paint" capability, we curate a high-quality dataset centered around quadruples of (T, R, L, I_{tgt}), representing the Prompt, Reasoning, Layout, and target Image, respectively. The construction process focuses on establishing a logical causal chain:

Logic-Driven Layout Annotation (L). Unlike existing datasets [51, 13] that rely on indiscriminate object detection, we aim to train the model to plan layouts that are *semantically deducible* from the text. We employ Qwen3-VL-235B-A22B [3] to analyze both the text prompt T and the target image I_{tgt}. We prompt the model to identify and bound only those objects required or implied by the prompt, filtering out irrelevant background. This ensures the layout L serves as a precise execution plan for the text.

Reasoning Chain Synthesis (R). To bridge the gap between the abstract prompt T and the concrete layout L, we synthesize the "Thinking" process. We prompt Qwen3-VL-235B-A22B to act as a spatial reasoner: given T and the annotated L, it generates a reasoning chain R that explicitly articulates the decomposition logic, spatial constraints, and attribute determinations that lead to the final layout. Additionally, we use an LLM to reject reasoning that is inconsistent with the layout or contains logical errors, ensuring high-quality supervision for the Think phase.

Human-Verified Quality Control. To mitigate bias from synthesized reasoning data and LLM-based filtering, we iteratively refine the VLM/LLM prompts through manual inspection and verify the final data quality with an author audit. Two authors check 500 layout annotations and confirm 95.3% grounding accuracy for main objects. They further check 200 synthesized reasoning chains and confirm 93.5% reasoning accuracy, judging whether each chain is logically consistent with the prompt and the annotated layout.

Data Diversity and Task Mixture. To ensure robustness and prevent the model from over-relying on the full reasoning pipeline, we incorporate diverse data formats:

- **"Think, Plan, Paint" Data** (T, R, L, I_{tgt})**:** For learning complex spatial reasoning.
- **Layout-Only Data** (T, L, I_{tgt})**:** Standard grounded generation data without explicit reasoning.
- **Image-Only Data** (T, I_{tgt})**:** Large-scale text-to-image pairs to maintain general visual synthesis quality.

Finally, we conduct **joint training** on a mixture of these generation samples along with layout-aware Multimodal Understanding data (I_{src}, T, L) and Image Editing Data (I_{src}, T, L, I_{tgt}).

3.3.2 Joint Optimization

We train ATLAS using a multi-task objective that combines language modeling and diffusion training. For the discrete component (reasoning and layout tokens), we minimize the standard negative log-likelihood loss $\mathcal{L}_{txt}$. For the continuous component (image latents), we employ the diffusion loss $\mathcal{L}_{diff}$ (mse on velocity prediction). The total objective is formulated as:

$$\mathcal{L} = \mathcal{L}_{txt}(R, L|T, I_{src}) + \lambda \mathcal{L}_{diff}(I_{tgt}|T, R, L, I_{src}) \tag{1}$$

The weight λ balances the two losses. We set λ following the training strategies of our base models.

Table 1: Quantitative comparison on T2I-CompBench and ATLAS-Reasoning. We report the scores across six dimensions for T2I-CompBench and the ATLAS-Reasoning benchmark. ATLAS demonstrates comprehensive superiority in complex spatial reasoning and non-spatial logic attributes. For each metric, higher is better. The best results are **bolded**.

Method	T2I-CompBench						ATLAS-Reasoning
	Color	Shape	Texture	Spatial	Non-Spa.	Numeracy	Spatial
General T2I Models							
SDXL [38]	69.44	64.11	79.90	68.33	89.16	48.33	47.21
SD v3.5 [12]	90.37	80.37	93.12	80.83	91.66	17.43	54.17
FluX.2-Dev [24]	**97.22**	90.74	97.61	97.56	82.53	**91.25**	89.72
Nano Banana Pro [10]	94.44	88.70	**97.81**	98.53	**98.33**	87.12	**95.31**
GPT-Image-1 [22]	95.74	**92.31**	97.78	94.17	96.67	85.35	93.48
Show-O2 [46]	84.44	77.41	81.96	77.50	92.53	69.16	51.27
BAGEL	91.11	81.64	93.12	78.33	90.50	70.83	52.01
BAGEL (Think)	89.42	79.63	92.81	81.95	91.31	67.14	58.94
Hunyuan Image 3.0	84.81	85.28	91.30	83.33	97.50	66.67	69.41
Hunyuan Image 3.0 (Think)	87.93	88.70	92.77	88.12	91.67	73.43	68.77
Layout Pipeline/Agents							
LayoutGPT [13]	34.16	47.13	43.10	34.41	46.59	47.93	21.98
GenArtist [43]	89.43	87.17	84.32	95.31	92.56	84.41	88.29
Layout-Based Unified MLLMs							
PlanGen [18]	67.41	61.28	77.00	46.67	74.17	40.83	25.43
ATLAS-7B	97.13	86.17	96.26	86.01	93.79	83.41	72.31
ATLAS-80B	96.42	90.16	90.62	**99.17**	96.67	85.08	91.21

3.3.3 RL-based Layout Alignment

After supervised joint training, we further refine the **Paint** stage with Flow-GRPO [29]. For each condition (T, L) or (T, R, L), the model samples a group of images and optimizes a group-relative objective. We use a weighted sum of five rewards: layout position, prompt-level semantic alignment, region counting, negative-region suppression, and image quality. For example, the layout-position reward evaluates whether generated objects appear in the specified boxes by matching detected objects with the input layout using semantic labels and spatial overlap. After matching, we compute it as an IoU-based F1 score, $r_{\text{pos}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall} + \epsilon}$. Detailed definitions are provided in Appendix D.3, and the contribution of this stage is evaluated in the ablation experiments.

4 Empirical Experiments

Implementation Details We instantiate ATLAS on BAGEL [11] and Hunyuan Image 3.0 [6], and we refer to these two variants as ATLAS-7B and ATLAS-80B, respectively. Both models are trained on a mixed dataset of 5 million samples, combining our curated reasoning-layout-image quadruples, layout-aware editing and understanding datasets, along with additional large-scale text-image pairs to maintain generalization. For ATLAS-7B, we set the batch size of 256 with a learning rate of 2.5e-5. For ATLAS-80B, we use a batch size of 1536 and a learning rate of 1e-4. For the RL-based alignment stage, we use Flow-GRPO with a learning rate of 1e-6, a group size of 16 images per prompt-layout condition, a clipping range of 0.2, and a KL coefficient of 0.01.

Our experiments aim to answer the following questions:

- **RQ1:** How does ATLAS perform on compositional text-to-image generation tasks compared to state-of-the-art baselines?
- **RQ2:** How effective is ATLAS at generating layouts that reflect complex spatial instructions?
- **RQ3:** Does ATLAS support precise user control through layout editing?
- **RQ4:** Can ATLAS effectively handle multimodal understanding and editing tasks?
- **RQ5:** What is the contribution of each key component of ATLAS to the overall performance?



Figure 4: Qualitative comparison of text-to-image generation. ATLAS better follows complex prompts involving object counts, attributes, and spatial relations.

Table 2: **Layout Planning Evaluation.** Comparison of generated layout quality against baselines. We report both spatial relation scores judged by Qwen3-VL-235B-A22B and rule-based clause checking accuracy on T2I-CompBench (Spatial) and ATLAS-Reasoning benchmarks.

Eval.	Dataset	PlanGen [18]	ATLAS-7B w/o Think	**ATLAS-7B**	ATLAS-80B w/o Think	**ATLAS-80B**
Qwen3-VL-235B-A22B	T2I-CompBench (Spatial)	37.31	67.31	82.30	83.68	**89.45**
	ATLAS-Reasoning	18.41	34.56	54.72	78.81	**84.29**
Rule-based	T2I-CompBench (Spatial)	29.84	54.63	72.96	73.12	**78.35**
	ATLAS-Reasoning	17.92	32.41	48.67	64.38	**73.92**

4.1 RQ1: Text-to-Image Generation Evaluation

Datasets. To comprehensively evaluate text-to-image generation capabilities, we utilize two benchmarks:

- **T2I-CompBench [21]:** A standardized benchmark for compositional generation, covering dimensions such as color binding, shape, and spatial relationships.
- **ATLAS-Reasoning:** To evaluate the reasoning capability of our CoT mechanism, we construct a challenging dataset comprising 500 complex prompts. These prompts require **multi-step reasoning** to solve, including entangled spatial relations (e.g., "A is left of B, and B is above C") or attribute binding (e.g., "There are four balloons, of which two are red and are not next to each other").

We follow the standard practice in T2I-CompBench to use MLLMs for evaluation. Specifically, we use the prompts provided by T2I-CompBench and employ Qwen3-VL-235B-A22B [3] to assess generation quality. For ATLAS-Reasoning, we follow the method for evaluating spatial relations in T2I-CompBench.

Results. We compare ATLAS against state-of-the-art general text-to-image models, layout-based generation pipelines, and layout-based unified MLLMs. As shown in Table 1, ATLAS-80B achieves results comparable to closed-source models such as Nano Banana Pro on T2I-CompBench, while outperforming layout-based pipelines. On the other hand, ATLAS-7B already surpasses existing layout-based unified MLLMs by 56.59% on average, demonstrating the effectiveness of our approach across different model scales. Compared with their corresponding base models, ATLAS-7B and ATLAS-80B achieve average relative improvements of 21.47% and 24.64%, respectively, on spatially related tasks. Especially on the challenging ATLAS-Reasoning benchmark, PlanGen only achieves 25.43, while the two scales of ATLAS still maintain strong reasoning capabilities.

Figure 4 presents qualitative comparisons. ATLAS outperforms competitors when the prompts involve complex spatial arrangements, or objects have various attributes. For example, when prompted with "six apples are lying on the ground. Only two apples are green." The thinking process of ATLAS correctly deduces the number and colors of apples, while the layout concretely specifies their positions and colors. In contrast, other models either miscount the apples or fail to assign the correct colors, leading to inaccurate generations.

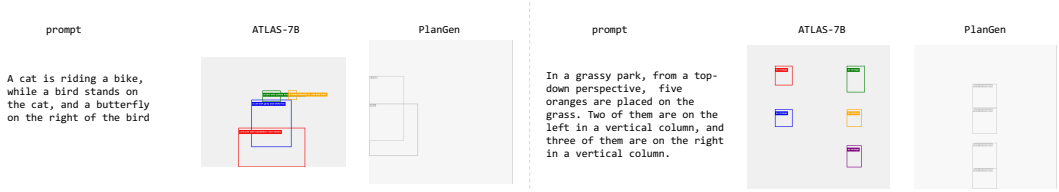


Figure 5: Qualitative comparison of layout planning. ATLAS more faithfully captures complex spatial constraints, including transitive relations, whereas PlanGen produces incomplete or inconsistent layouts.

Table 3: **Layout Shift Consistency on T2I-CompBench.** We report the Hit Rate (HR@0.5) under the challenging Adjacency Shift setting using Qwen3-VL-235B-A22B and GroundingDINO as detectors. A higher score indicates the model can precisely generate objects at new, user-specified locations without positional inertia.

Detector	PlanGen [18]	**ATLAS-7B**
Qwen3-VL-235B-A22B	79.69	**90.17**
GroundingDINO	77.34	**87.66**

4.2 RQ2: Layout Planning Evaluation

To assess layout planning capabilities of ATLAS, we evaluate the quality of generated bounding boxes given text prompts. We compare ATLAS against PlanGen and ATLAS without the **"Think"** phase (i.e., directly generating layouts from text). We use the ATLAS-Reasoning dataset and the spatial subset of T2I-CompBench for evaluation. We evaluate layouts with Qwen3-VL-235B-A22B [3] and a rule-based clause checker, measuring whether each object is present and each spatial relation is satisfied.

Results. Table 2 summarizes the results under both evaluators. ATLAS outperforms PlanGen by a large margin, and the inclusion of the **"Think"** phase significantly boosts layout planning accuracy. This confirms that the "Think" phase effectively enhances the model's reasoning about spatial relationships before layout generation. Under Qwen3-VL-235B-A22B evaluation, the "Think" phase brings larger improvements on the smaller model (7B), improving the score by an average of 17.6, compared with 5.62 on the 80B model. Rule-based checking shows the same trend. We attribute this to the fact that the larger model's ability is already strong enough to handle most spatial arrangements without explicit CoT.

Figure 5 shows qualitative examples. The "Think" phase enables ATLAS to generate layouts that accurately reflect complex spatial instructions, while PlanGen often produces layouts with missing objects, incorrect quantifiers, and spatial relations.

4.3 RQ3: Precision in Controllable Generation

To address RQ3, we evaluate the fidelity of the "Plan $\rightarrow$ Paint" phase. In real-world interactive applications, users typically utilize bounding boxes as precise control handles to refine generated results. Therefore, a truly controllable model must be responsive to these manual adjustments, strictly adhering to the new spatial constraints rather than relying on internal priors or "positional inertia" from the initial generation.

Table 4: Ablation studies on ATLAS.

Method	**T2I-CompBench**					
	Color	Shape	Texture	Spatial	Non-Spa.	Numeracy
Full Model (ATLAS-7B)	97.13	**86.17**	**96.26**	**86.01**	93.79	**83.41**
w/o RL	**97.41**	83.31	93.35	85.83	93.33	80.83
w/o Shared Tokens (Plan with text and number)	95.13	79.70	94.27	81.16	**94.81**	73.02
w/o MMU data	91.29	79.90	88.81	79.83	91.67	65.83
w/o Think	97.29	82.30	93.27	75.17	93.19	62.33
w/o Plan	92.42	80.16	90.62	80.83	91.67	61.08

Table 5: Evaluation results on Instruction-Guided Editing.

Method	ImgEdit-Bench									
	Add	**Adjust**	**Extract**	**Replace**	**Remove**	**Background**	**Style**	**Hybrid**	**Action**	**Overall**
BAGEL-7B [11]	3.56	3.31	1.88	2.62	2.88	3.44	4.49	2.38	4.17	3.20
ATLAS-7B	**4.04**	3.33	1.87	**3.98**	**3.21**	**3.57**	4.41	2.41	**4.34**	**3.46**

Table 6: Evaluation results on Multimodal Understanding. We report RefCOCO, MMMU, and POPE scores.

	RefCOCO	MMMU	POPE
Qwen2.5-VL-7B [4]	**91.7**	53.1	**90.2**
ATLAS-7B	90.5	**54.7**	88.1

We simulate this interactive refinement scenario using the T2I-CompBench dataset. Specifically, we conduct a controlled Layout Shift Test on valid layouts where the target object is correctly generated and matches the original box. For this target object, we apply an Adjacency Shift perturbation: we spatially translate the bounding box until it is disjoint from but adjacent to its original position (sharing only a border, resulting in an IoU ≈ 0). This setting represents a challenging limit case, as it forces the model to synthesize the object in a completely fresh region while effectively "erasing" it from the original coordinates.

We regenerate the image conditioned on the *shifted* layout and detect the target object with Qwen3-VL-235B-A22B and GroundingDINO [34]. We report the Hit Rate (HR@0.5), which counts a generation as successful if the detected object overlaps with the *new* target box with an IoU > 0.5. This metric strictly measures the execution precision of the spatial adjustment.

Results. As shown in Table 3, ATLAS achieves the highest success rate under both detectors. While the baseline method often struggles to relocate objects away from their initial or canonical positions (positional inertia), our method effectively adheres to the updated layout constraints, validating its superior controllability for interactive fine-grained layout adjustments.

4.4 RQ 4: Versatility of the Unified Framework

To address **RQ4**, we demonstrate that ATLAS is not merely a generator, but a unified system capable of precise editing and multimodal understanding.

Instruction-Guided Editing. We evaluate the editing capabilities on the ImgEdit-Bench [47] dataset. We compare ATLAS-7B against its base model, BAGEL-7B, to verify the contribution of our unified planning mechanism. As shown in Table 5, ATLAS achieves consistent improvements over BAGEL on average. Especially on Add, Replace, and Remove tasks, ATLAS achieves significant improvements. This indicates that explicitly modeling the "Plan" phase allows for more precise structural manipulation.

Multimodal Understanding. We further evaluate multimodal understanding on RefCOCO grounding, MMMU [50], and POPE [25], comparing ATLAS-7B with Qwen2.5-VL-7B [4]. As shown in Table 6, ATLAS-7B remains competitive on RefCOCO and POPE, while slightly outperforming Qwen2.5-VL-7B on MMMU.

4.5 RQ 5: Ablation Studies

To answer **RQ5** and quantify the contribution of each core component, we conduct ablation studies on the T2I-CompBench benchmark. We train several variants of ATLAS-7B by selectively removing key modules or training strategies: RL-based layout alignment, shared positional tokens, multimodal understanding data, the **"Think"** phase, and the **"Plan"** phase. As shown in Table 4, removing these components leads to performance drops, especially on spatial and numeracy dimensions, which highlights the importance of these designs in aligning text and vision modalities for precise spatial reasoning and layout planning. Additionally, the results discussed in Section 4.2 can further validate the effectiveness of the "Think" phase.

5 Conclusion

In this paper, we presented ATLAS, a unified native MLLM framework that redefines controllable image synthesis through a human-like "Think, Plan, Paint" paradigm. By introducing a Layout-Aware Chain-of-Thought mechanism and Shared Positional Tokens, ATLAS effectively bridges the semantic gap between abstract textual reasoning and concrete visual planning. Extensive evaluations across 7B and 80B scales demonstrate that our approach not only establishes new state-of-the-art results on complex compositional generation benchmarks but also achieves versatile performance in instruction-guided editing and visual grounding. Ultimately, ATLAS offers a scalable, transparent, and robust path toward fine-grained controllable image generation in unified multimodal models.

Acknowledgments and Disclosure of Funding

Funding in direct support of this work. This work was conducted during Junhao Liu's internship at Tencent Hunyuan, which provided the computational resources used in this research.

Competing interests. The authors declare no competing interests beyond the affiliations and support disclosed above.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. URL https://arxiv.org/abs/2204.14198.

[2] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

[3] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

[5] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. URL https://arxiv.org/abs/2211.09800.

[6] Siyu Cao, Hangting Chen, Peng Chen, Yiji Cheng, Yutao Cui, Xinchi Deng, Ying Dong, Kipper Gong, Tianpeng Gu, Xiusen Gu, Tiankai Hang, Duojun Huang, Jie Jiang, Zhengkai Jiang, Weijie Kong, Changlin Li, Donghao Li, Junzhe Li, Xin Li, Yang Li, Zhenxi Li, Zhimin Li, Jiaxin Lin, Linus, Lucaz Liu, Shu Liu, Songtao Liu, Yu Liu, Yuhong Liu, Yanxin Long, Fanbin Lu, Qinglin Lu, Yuyang Peng, Yuanbo Peng, Xiangwei Shen, Yixuan Shi, Jiale Tao, Yangyu Tao, Qi Tian, Pengfei Wan, Chunyu Wang, Kai Wang, Lei Wang, Linqing Wang, Lucas Wang, Qixun Wang, Weiyan Wang, Hao Wen, Bing Wu, Jianbing Wu, Yue Wu, Senhao Xie, Fang

Yang, Miles Yang, Xiaofeng Yang, Xuan Yang, Zhantao Yang, Jingmiao Yu, Zheng Yuan, Chao Zhang, Jian-Wei Zhang, Peizhen Zhang, Shi-Xue Zhang, Tao Zhang, Weigang Zhang, Yepeng Zhang, Yingfang Zhang, Zihao Zhang, Zijian Zhang, Penghao Zhao, Zhiyuan Zhao, Xuefei Zhe, Jianchen Zhu, and Zhao Zhong. Hunyuanimage 3.0 technical report, 2025. URL https://arxiv.org/abs/2509.23951. Licensed under the Tencent Hunyuan Community License.

[7] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5343–5353, January 2024.

[8] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

[9] Jaemin Cho, Abhay Zala, and Mohit Bansal. Visual programming for text-to-image generation and evaluation, 2023. URL https://arxiv.org/abs/2305.15328.

[10] Gheorghe Comanici and et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

[11] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining, 2025. URL https://arxiv.org/abs/2505.14683. Licensed under the Apache License 2.0.

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. URL https://arxiv.org/abs/2403.03206.

[13] Weixi Feng, Wanrong Zhu, Tsu jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models, 2023. URL https://arxiv.org/abs/2305.15393.

[14] Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models, 2024. URL https://arxiv.org/abs/2309.17102.

[15] Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. Seed-x: Multimodal models with unified multi-granularity comprehension and generation, 2025. URL https://arxiv.org/abs/2404.14396.

[16] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 52132–52152. Curran Associates, Inc., 2023. doi: 10.52202/075280-2270. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a3bf71c7c63f0c3bcb7ff67c67b1e7b1-Paper-Datasets_and_Benchmarks.pdf.

[17] Junwen He, Yifan Wang, Lijun Wang, Huchuan Lu, Jun-Yan He, Chenyang Li, Hanyuan Chen, Jin-Peng Lan, Bin Luo, and Yifeng Geng. Gldesigner: Leveraging multi-modal llms as designer for enhanced aesthetic text glyph layouts, 2025. URL https://arxiv.org/abs/2411.11435.

[18] Runze He, Bo Cheng, Yuhang Ma, Qingxiang Jia, Shanyuan Liu, Ao Ma, Xiaoyu Wu, Liebucha Wu, Dawei Leng, and Yuhui Yin. Plangen: Towards unified layout planning and image generation in auto-regressive vision language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18143–18154, 2025.

[19] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022. URL https://arxiv.org/abs/2208.01626.

[20] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment, 2024. URL https://arxiv.org/abs/2403.05135.

[21] Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation, 2025. URL https://arxiv.org/abs/2307.06350.

[22] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[23] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1086. URL https://aclanthology.org/D14-1086/.

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

[25] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://openreview.net/forum?id=xozJw0kZXF.

[26] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22511–22521, 2023.

[27] Dongyang Liu, Shitian Zhao, Le Zhuo, Weifeng Lin, Yi Xin, Xinyue Li, Qi Qin, Yu Qiao, Hongsheng Li, and Peng Gao. Lumina-mgpt: Illuminate flexible photorealistic text-to-image generation with multimodal generative pretraining, 2025. URL https://arxiv.org/abs/2408.02657.

[28] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. URL https://arxiv.org/abs/2304.08485.

[29] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl, 2025. URL https://arxiv.org/abs/2505.05470.

[30] Junhao Liu and Xin Zhang. Rex: A framework for incorporating temporal information in model-agnostic local explanation techniques. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(18):18888–18896, Apr. 2025. doi: 10.1609/aaai.v39i18.34079. URL https://ojs.aaai.org/index.php/AAAI/article/view/34079.

[31] Junhao Liu, Haonan Yu, and Xin Zhang. Conlux: Concept-based local unified explanations. *arXiv preprint arXiv:2410.12439*, 2024.

[32] Junhao Liu, Haonan Yu, Zhenyu Yan, and Xin Zhang. Focus-lime: Surgical interpretation of long-context large language models via proxy-based neighborhood selection. In Diego Calvanese, editor, *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence, IJCAI-26*, pages 4572–4580. International Joint Conferences on Artificial Intelligence Organization, 8 2026. doi: 10.24963/ijcai.2026/509. URL https://doi.org/10.24963/ijcai.2026/509. Main Track.

[33] Junhao Liu, Haonan Yu, Zhenyu Yan, and Xin Zhang. Revitalizing black-box interpretability: Actionable interpretability for LLMs via proxy models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4806–4844, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.220. URL https://aclanthology.org/2026.acl-long.220/.

[34] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[35] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. Deepseek-vl: Towards real-world vision-language understanding, 2024. URL https://arxiv.org/abs/2403.05525.

[36] Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. Chameleon: Plug-and-play compositional reasoning with large language models. *Advances in Neural Information Processing Systems*, 36:43447–43478, 2023.

[37] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2302.08453.

[38] 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, 2023. URL https://arxiv.org/abs/2307.01952.

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

[40] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

[41] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1527–1535. AAAI Press, 2018.

[42] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Emu: Generative pretraining in multimodality, 2024. URL https://arxiv.org/abs/2307.05222.

[43] Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. Genartist: Multimodal llm as an agent for unified image generation and editing, 2024. URL https://arxiv.org/abs/2407.05600.

[44] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7452–7461, October 2023.

[45] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2408.12528.

[46] Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025.

[47] Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025.

[48] Haonan Yu, Junhao Liu, Zhenyu Yan, Haoran Lin, and Xin Zhang. Wasd: Locating critical neurons as sufficient conditions for explaining and controlling llm behavior, 2026. URL https://arxiv.org/abs/2603.18474.

[49] Haonan Yu, Junhao Liu, and Xin Zhang. MAnchors: Memorization-based acceleration of anchors via rule reuse and transformation. In Tong Zhang, Miroslav Dudik, Martin Jaggi, Alekh Agarwal, Sharon Li, Dale Schuurmans, Jerry Zhu, Felix Berkenkamp, Hanze Dong, and Alberto Bietti, editors, *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 150722–150735. PMLR, 06–11 Jul 2026. URL https://proceedings.mlr.press/v306/yu26ae.html.

[50] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi, 2024. URL https://arxiv.org/abs/2311.16502.

[51] Hui Zhang, Dexiang Hong, Tingwei Gao, Yitong Wang, Jie Shao, Xinglong Wu, Zuxuan Wu, and Yu-Gang Jiang. Creatilayout: Siamese multimodal diffusion transformer for creative layout-to-image generation. *arXiv preprint arXiv:2412.03859*, 2024.

[52] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2302.05543.

[53] Yuyao Zhang, Jinghao Li, and Yu-Wing Tai. Layercraft: Enhancing text-to-image generation with cot reasoning and layered object integration, 2025. URL https://arxiv.org/abs/2504.00010.

[54] Wang Zhenyu, Xie Enze, Li Aoxue, Wang Zhongdao, Liu Xihui, and Li Zhenguo. Divide and conquer: Language models can plan and self-correct for compositional text-to-image generation. *arXiv preprint arXiv:2401.15688*, 2024.

[55] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024.

A LLM Usage Statements

We used LLMs to polish our writing during paper preparation.

B Limitations

This work uses bounding boxes as the layout interface. This representation is simple, interpretable, and well aligned with existing grounding and layout-aware generation benchmarks, which makes it suitable for studying the alignment between textual reasoning, layout planning, and visual generation.

At the same time, bounding boxes are only one possible way to represent layouts. Other interfaces may better capture certain types of visual structure or user intent, especially in scenarios that require more detailed spatial control. Exploring how different layout interfaces affect reasoning, planning, and generation quality is an important direction for future work.

C Broader Impacts

This work aims to improve controllable image generation by enabling models to better follow explicit layout instructions. Such capability can benefit creative design, visual communication, education, and accessibility-oriented content creation, where users may need more precise control over object placement and spatial relationships.

At the same time, more controllable image generation can also lower the barrier to producing misleading or harmful visual content. Potential misuse includes generating deceptive media, manipulating visual narratives, or creating biased and inappropriate images. We encourage responsible deployment with content provenance, safety filtering, and human oversight, especially in applications that may affect public information, identity, or vulnerable groups.

To mitigate these risks, we exclude harmful or unsafe content from the training data used in this work as a data-level safeguard.

D Experimental Details

D.1 Compute Resources

Our experiments were conducted on a GPU cluster with 256 GPUs in total, where each GPU is equipped with 96GB of memory.

D.2 Statistical Significance Tests

For the main text-to-image generation results, we conduct paired significance tests over prompt-level evaluation scores. We compare ATLAS with PlanGen and with the corresponding base model, i.e., BAGEL for ATLAS-7B and Hunyuan Image 3.0 for ATLAS-80B. Using paired bootstrap resampling, we test whether the mean prompt-level score difference is greater than zero. At the significance level $\alpha = 0.01$, the improvements of ATLAS remain statistically significant in both comparison groups.

D.3 Reward Details for RL-based Layout Alignment

We use five rewards in the RL-based layout alignment stage and combine them with a weighted sum:

$$r(I, T, L) = w_{\text{pos}} r_{\text{pos}} + w_{\text{sem}} r_{\text{sem}} + w_{\text{cnt}} r_{\text{cnt}} + w_{\text{neg}} r_{\text{neg}} + w_{\text{qua}} r_{\text{qua}}. \tag{2}$$

Layout-position reward. We estimate object boxes from the generated image and match them with the input layout boxes using semantic labels and spatial overlap. The reward is computed as an IoU-based F1 score:

$$r_{\text{pos}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall} + \epsilon}. \tag{3}$$

Region-counting reward. For a layout with N regions, let c_j be the expected number of target objects in the j-th region, and let $\hat{c}_j$ be the estimated count from the generated image. We define:

$$r_{\text{cnt}} = \frac{1}{N} \sum_{j=1}^{N} \max\left(0, 1 - \frac{|\hat{c}_j - c_j|}{\max(c_j, 1)}\right). \tag{4}$$

Semantic-alignment reward. To preserve global prompt fidelity, we compute the CLIP similarity between the text prompt and the generated image and normalize it into a bounded reward:

$$r_{\text{sem}} = \sigma\left(\tau \cdot \cos\left(E_{\text{text}}(T), E_{\text{image}}(I)\right)\right), \tag{5}$$

where E_{text} and E_{image} are the text and image encoders, and τ is a temperature.

Negative-region reward. This reward penalizes target objects that appear outside the specified layout regions. Given the number of unmatched detected target objects $N_{\text{unmatched}}$ and all detected target objects N_{pred}, we define:

$$r_{\text{neg}} = 1 - \frac{N_{\text{unmatched}}}{\max(N_{\text{pred}}, 1)}. \tag{6}$$

Image-quality reward. Finally, we use an image preference model to encourage visual realism and aesthetics:

$$r_{\text{qua}} = \sigma\left(s_{\text{quality}}(I)\right). \tag{7}$$

All reward terms are normalized to comparable ranges before weighted aggregation.

E T2I CompBench Evaluation Details

In our evaluation pipeline, we utilize MLLMs to assess the alignment between generated images and text prompts. Below, we detail the specific instruction templates used for each category in T2I-CompBench. In these templates, placeholders such as `[Category]`, `[Object]`, and `[Prompt]` are dynamically replaced with the specific attributes, objects, or captions from the dataset.

E.1 Attribute Binding (Color, Shape, Texture)

For attribute binding tasks, we evaluate whether a specific object possesses the correct attribute. The scoring range is 1-4.

System Instruction: You are my assistant to identify any objects and their `[Category]` in the image. According to the image, evaluate if there is a `[Object]` in the image.
Criteria:
- **4**: There is `[Object]`, and `[Category]` is `[Attribute]`.
- **3**: There is `[Object]`, `[Category]` is mostly `[Attribute]`.
- **2**: There is `[Object]`, but it is not `[Attribute]`.
- **1**: No `[Object]` in the image.

Output Format: Provide your analysis and explanation in JSON format with the following keys: score (range 1-4, e.g., 1), explanation (within 20 words).

E.2 Spatial Relationships (Spatial, 3D, Shape-Multistep)

For spatial categories, we assess the layout accuracy. The scoring range is 1-5.

System Instruction: You are my assistant to identify objects and their spatial layout in the image. According to the image, evaluate if the text "`[Prompt Text]`" is correctly portrayed in the image.
Criteria:
- **5**: Correct spatial layout in the image for all objects mentioned in the text.
- **4**: Basically, spatial layout of objects matches the text.
- **3**: Spatial layout not aligned properly with the text.
- **2**: Image not aligned properly with the text.
- **1**: Image almost irrelevant to the text.

Output Format: Provide your analysis and explanation in JSON format with the following keys: score (range 1-5, e.g., 2), explanation (within 50 words).

E.3 Action and Non-Spatial Relationships

This category evaluates actions, events, and inter-object relationships. The scoring range is 1-5.

System Instruction: You are my assistant to identify the actions, events, objects and their relationships in the image. According to the image, evaluate if the text "`[Prompt Text]`" is correctly portrayed in the image.
Criteria:
- **5**: The image accurately portrayed the actions, events and relationships between objects described in the text.
- **4**: The image portrayed most of the actions, events and relationships but with minor discrepancies.
- **3**: The image depicted some elements, but action relationships between objects are not correct.
- **2**: The image failed to convey the full scope of the text.
- **1**: The image did not depict any actions or events that match the text.

Output Format: Provide your analysis and explanation in JSON format with the following keys: score (range 1-5, e.g., 2), explanation (within 20 words).

E.4 Numeracy

This category evaluates the correct quantity of objects. The scoring range is 1-5.

System Instruction: You are my assistant to identify objects and their quantities in the image. According to the image and your previous answer, evaluate how well the image aligns with the text prompt: "`[Prompt Text]`".
Criteria:
- **5**: Correct numerical content in the image for all objects mentioned in the text.
- **4**: Basically, numerical content of objects matches the text.
- **3**: Numerical content not aligned properly with the text.
- **2**: Image not aligned properly with the text.
- **1**: Image almost irrelevant to the text.

Output Format: Provide your analysis and explanation in JSON format with the following keys: score (range 1-5, e.g., 2), explanation (within 20 words).

E.5 Additional Results

E.5.1 Editing Examples

We further present editing examples in Figure 6, showcasing ATLAS's ability to adapt to user-modified layouts and generate images that accurately reflect these changes. Notably, we do not need to manually use negative prompts to prevent the presence of removed objects, as ATLAS inherently understands and adheres to the updated layout instructions.



Figure 6: Additional editing examples using ATLAS.

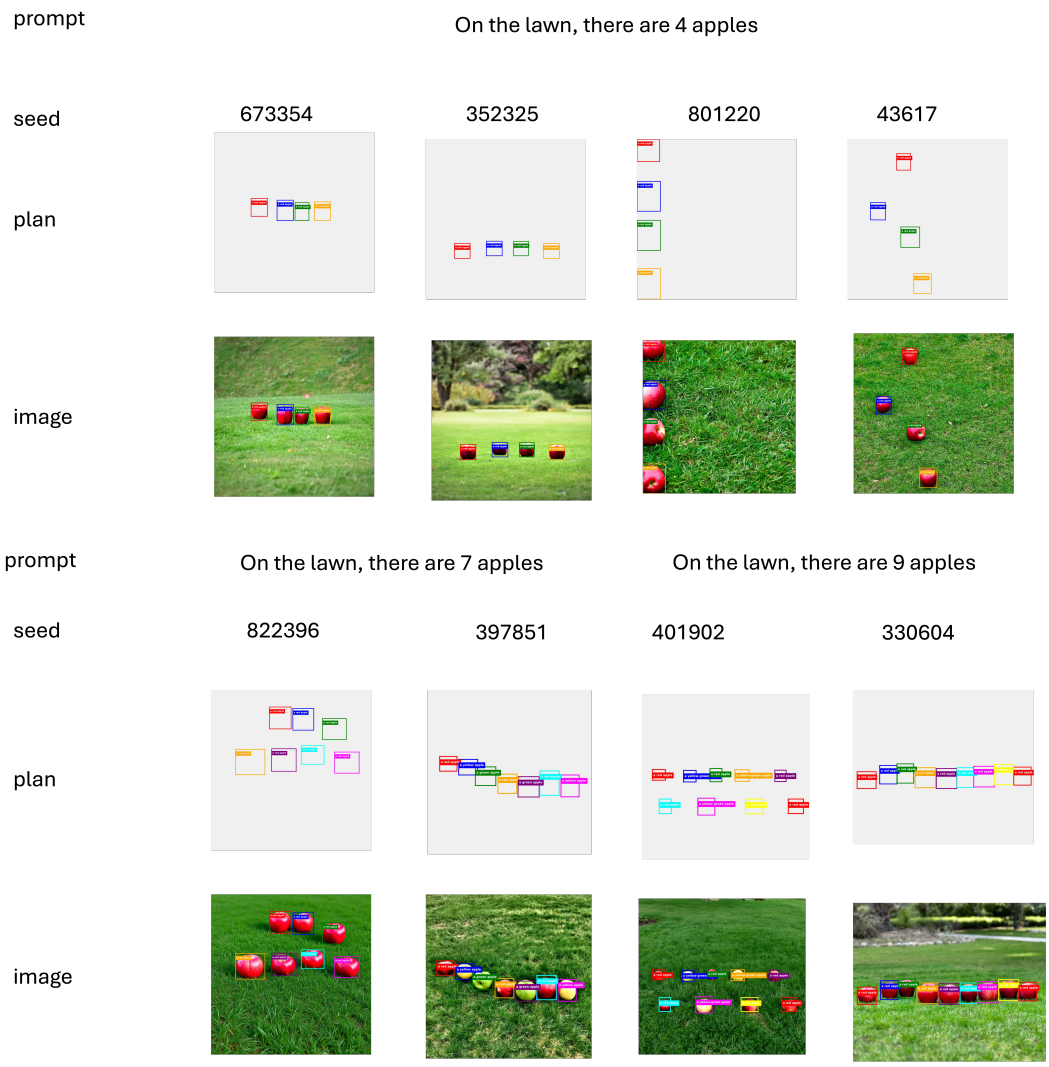


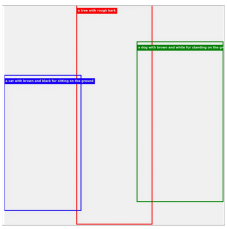
Figure 7: Layout diversity under different random seeds. For the same counting prompts, ATLAS samples multiple valid layouts with distinct object positions and optional attributes, while preserving the target object counts in the generated images.

E.5.2 Layout Diversity

A potential concern for layout-based generation is that the intermediate layout representation may collapse to a small set of fixed spatial templates. ATLAS avoids this issue because both the Think and Plan stages are generated autoregressively by the underlying LLM rather than retrieved from deterministic templates. As a result, standard stochastic decoding naturally induces a distribution over plausible layouts conditioned on the same text prompt. As shown in Figure 7, we qualitatively examine this property by sampling multiple layouts for counting-specific prompts with different random seeds, including prompts that require 4, 7, and 9 target objects. Across these samples, ATLAS preserves the requested object counts while producing diverse spatial arrangements and corresponding generated images. When the prompt leaves certain attributes unspecified, the sampled layouts can also vary optional object attributes such as color. This suggests that explicit layout planning provides controllability without sacrificing diversity, allowing the model to represent multiple valid visual realizations for the same instruction.

Think: "The cat is on the left, so its position should be anchored around <pos_x_200>. The dog is on the right, so it should be placed around <pos_x_700>. The tree is central and between them, around <pos_x_500>."

Plan:



a tree with rough bark: (331, 0) - (674, 999)
a cat with brown and black fur sitting on the ground: (0, 321) - (350, 936)
a dog with brown and white fur standing on the ground: (606, 168) - (999, 896)

Figure 8: Handling ambiguous spatial language. For prompts such as “a cat and a dog around a tree,” ATLAS first produces a reasonable default layout that satisfies the underspecified relation.

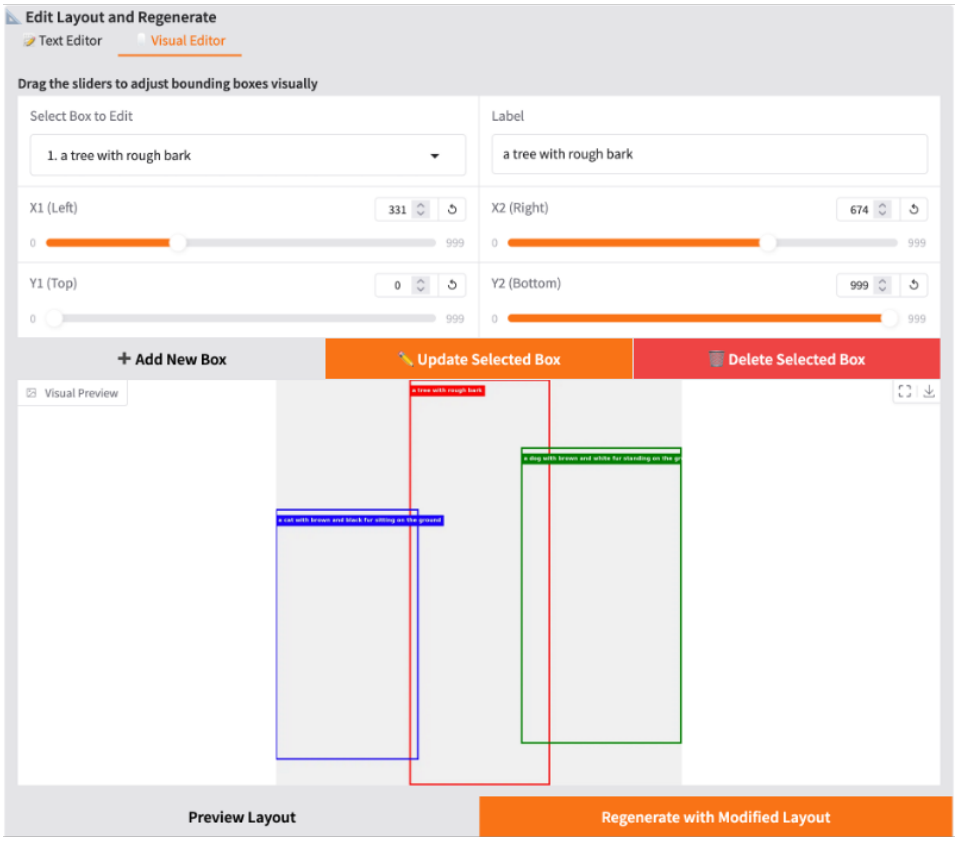


Figure 9: Interactive plan editing interface. Users can adjust, add, or delete intermediate bounding boxes before regeneration, which allows ambiguous spatial instructions to be refined into a user-specified arrangement.

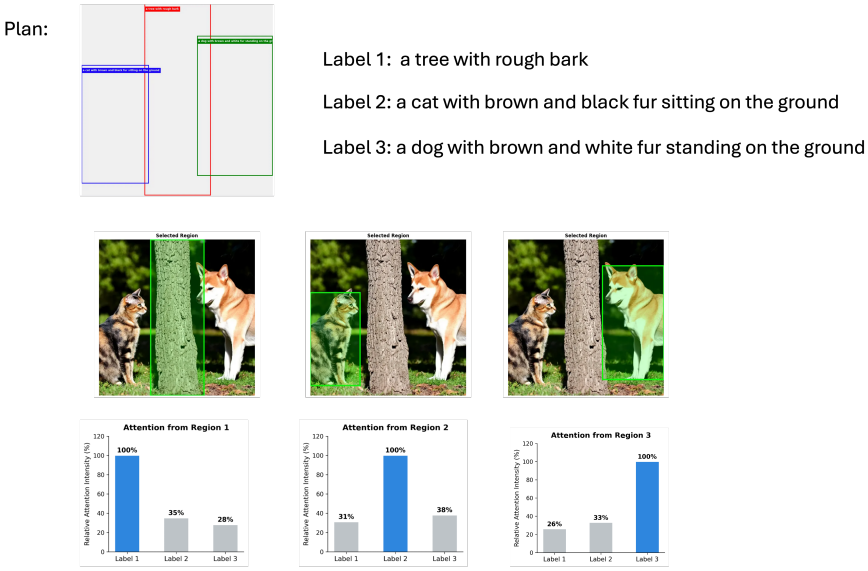


Figure 10: Attention analysis between layout planning and image generation. For each selected generated region, we visualize the attention from image tokens back to object labels in the layout plan. The generated regions attend most strongly to their corresponding layout labels, indicating that the Paint stage follows the object-level plan produced by the reasoning stage.

E.5.3 Ambiguous Spatial Language

Natural language spatial terms such as "near" and "around" are inherently ambiguous, since they often correspond to multiple valid object arrangements rather than a single canonical layout. ATLAS handles such prompts by combining natural-language reasoning with an editable intermediate plan. As shown in Figure 8, for the prompt "a cat and a dog around a tree," the Think stage explicitly reasons about the spatial relation and the Plan stage instantiates a physically and compositionally reasonable default layout. This default should be viewed as one plausible interpretation of the underspecified prompt, not as the only valid arrangement. When users have a more specific configuration in mind, they can directly modify the intermediate coordinates before generation, as illustrated in Figure 9. This turns ambiguity into an interactive control opportunity: ATLAS provides a reasonable initial plan, while the user can refine the layout before the final Paint stage.

E.5.4 Reasoning–Generation Interaction

The Paint stage is strictly conditioned on the output sequence produced by the preceding Think and Plan stages. To illustrate how the layout plan guides image synthesis, we analyze attention weights from generated image regions back to the object labels in the layout plan. As shown in Figure 10, each selected region in the generated image attends most strongly to its corresponding object label in the plan. For example, image tokens in the cat, tree, and dog regions assign the highest relative attention to the matching layout labels rather than unrelated objects. This indicates that the generation process does not merely use the layout as a loose global hint; instead, it maintains object-level correspondence between planned regions and generated visual content, helping the model focus on the areas specified by the intermediate plan.

E.5.5 Beyond Spatial Reasoning

Although ATLAS uses layouts as the intermediate representation, the reasoning performed before layout generation is not limited to explicit spatial relations. Because the Think stage is generated in the text modality, it can incorporate broader commonsense knowledge, including physical and causal implications that are only implicit in the prompt. Figure 11 shows an example with the prompt "a cup fell off the table." Rather than placing the cup on the tabletop, ATLAS infers that the cup should appear on the ground after falling, and the subsequent Plan and Paint stages realize this physically plausible scene. This example illustrates that layout planning can serve as an interface for

Prompt: On the lawn, a table is on the left of the image, and a cup fell off the table

Think: ...The cup fell off the table, thus should be on the ground

Plan:

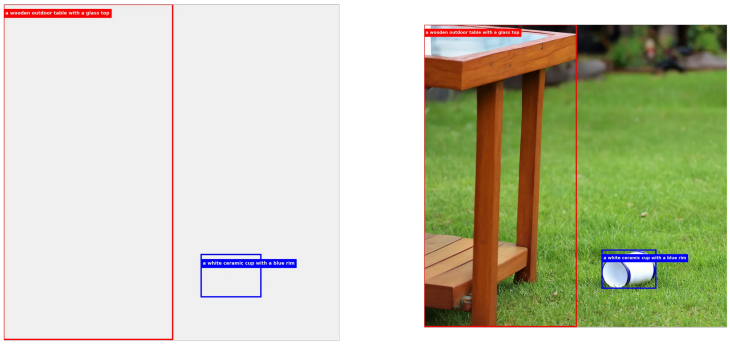


Figure 11: Example of physical reasoning in layout planning. Given a prompt implying that a cup has fallen off a table, ATLAS infers in the Think stage that the cup should be placed on the ground rather than on the tabletop, and the generated image follows this physically plausible layout.

Table 7: DPG-Bench evaluation results.

Model	Global	Entity	Attribute	Relation	Other	Overall↑
SDXL	83.27	82.43	80.91	86.76	80.41	74.65
BAGEL	88.94	90.37	91.29	90.82	88.67	85.07
BAGEL (Think)	87.29	88.67	91.30	91.67	87.11	85.92
Hunyuan Image 3.0	90.17	89.36	90.53	88.42	89.21	86.02
Hunyuan Image 3.0 (Think)	89.62	90.01	90.28	91.96	89.72	86.78
PlanGen	87.58	88.63	88.17	91.30	88.30	85.63
ATLAS-7B	**90.41**	90.13	**92.51**	92.75	89.54	87.29
ATLAS-80B	90.13	**91.36**	92.43	**94.42**	**90.32**	**88.20**

grounding not only spatial instructions, but also higher-level reasoning about events and their visual consequences.

E.5.6 Text-to-Image Generation Benchmarks

DPG-Bench [20]. Table 7 reports the detailed DPG-Bench results, which evaluate prompt following from multiple perspectives including global consistency, entity coverage, attribute binding, and relations. ATLAS achieves the best overall performance among all compared methods, with ATLAS-80B reaching an overall score of 88.20 and ATLAS-7B reaching 87.29. Compared with the corresponding base models, the two ATLAS variants consistently improve overall prompt-following accuracy, demonstrating that explicit layout-aware reasoning benefits not only spatial control but also general compositional alignment. The gains are especially clear on relation-centric and attribute-centric dimensions: ATLAS-80B obtains the best relation score of 94.42, while ATLAS-7B achieves the best attribute score of 92.51. These results suggest that the Think–Plan–Paint formulation helps the model bind objects, attributes, and relations into a coherent intermediate layout before image synthesis.

GenEval [16]. Table 8 further evaluates object-level compositional generation on GenEval. ATLAS-80B achieves the highest overall score of 0.92, and ATLAS-7B also reaches a strong overall score of 0.90, outperforming both their base models and the layout-planning baseline. The improvement

Table 8: GenEval evaluation results.

Model	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall↑
SDXL	0.98	0.74	0.39	0.85	0.15	0.23	0.55
BAGEL	0.98	0.95	0.84	**0.95**	0.78	**0.77**	0.88
Hunyuan Image 3.0	**1.00**	0.93	0.85	0.91	0.77	0.68	0.86
PlanGen	0.98	0.81	0.51	0.89	0.66	0.59	0.74
ATLAS-7B	0.98	**0.96**	0.88	**0.95**	0.85	**0.77**	0.90
ATLAS-80B	0.99	**0.96**	**0.92**	**0.95**	**0.91**	0.76	**0.92**

Table 9: Runtime comparison of different models. Runtime is measured in seconds per image.

Model	Runtime (s)
BAGEL	87.3
BAGEL (Think)	98.7
ATLAS-7B	99.4
Hunyuan Image 3.0	143.9
Hunyuan Image 3.0 (Think)	155.2
ATLAS-80B	156.5

is most pronounced on categories that require precise structural grounding, such as counting and position. For example, ATLAS-80B improves the position score from 0.77 to 0.91 over Hunyuan Image 3.0, while ATLAS-7B improves the position score from 0.78 to 0.85 over BAGEL. The consistent gains on counting and spatial placement indicate that explicit layout planning provides a reliable bridge between textual compositional constraints and visual realization.

E.5.7 Efficiency and Evaluation Details

Runtime. Table 9 reports the inference latency of ATLAS and the corresponding base models, measured in seconds per image under the same evaluation setting. ATLAS introduces only a modest computational overhead over the base generators. For the 7B model, ATLAS-7B takes 99.4 seconds per image, which is close to BAGEL with the Think stage enabled (98.7 seconds) and about 13.9% slower than the original BAGEL model without thinking. For the 80B model, ATLAS-80B takes 156.5 seconds per image, similarly matching Hunyuan Image 3.0 with thinking (155.2 seconds) and adding about 8.8% latency over the original Hunyuan Image 3.0. These results indicate that the explicit Think–Plan–Paint process substantially improves controllability and compositional alignment while keeping the inference cost comparable to base models with reasoning enabled.

In interactive scenarios, the thinking stage does not need to be treated as an opaque blocking step. As shown in Figure 12, our interface streams the model's thinking process while it reasons about the prompt and prepares the layout plan. Users can inspect this partial reasoning in real time and interrupt

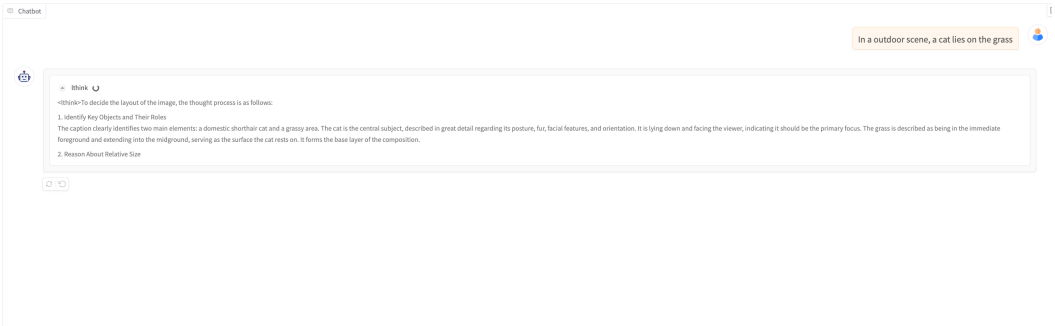


Figure 12: Interactive interface for ATLAS. The interface streams the model's thinking process to the user and supports interruption during reasoning, enabling more transparent and controllable interaction.

the process at any moment, for example when the streamed reasoning already reveals an undesired interpretation or when they want to revise the prompt before image synthesis. This design improves practical responsiveness: although the full Think–Plan–Paint pipeline has a modest overhead in end-to-end latency, users are not forced to wait for the entire thinking process when early feedback is sufficient.

F ATLAS-Reasoning Dataset

ATLAS-Reasoning contains 500 prompts designed to evaluate multi-step spatial reasoning and compositional image generation. The official repository for the benchmark and its accompanying documentation is `https://github.com/outerform/ATLAS-Reasoning`.

G Used Existing Assets and Licenses

- **BAGEL** [11]: base model for ATLAS-7B; Apache License 2.0.
- **HunyuanImage 3.0** [6]: base model for ATLAS-80B; Tencent Hunyuan Community License.
- **Qwen3-VL-235B-A22B** [3]: data annotation and automated evaluation; Apache License 2.0.
- **Qwen2.5-VL-7B**: multimodal-understanding baseline; Apache License 2.0.
- **OpenAI CLIP** [39]: semantic-alignment reward; MIT License.
- **GroundingDINO** [34]: detection-based evaluation; Apache License 2.0.
- **SDXL** [38]: text-to-image baseline; CreativeML Open RAIL++-M License.
- **Stable Diffusion 3.5** [12]: text-to-image baseline; Stability AI Community License.
- **FLUX.2 [dev]** [24]: text-to-image baseline; FLUX Non-Commercial License.
- **Show-O2** [46]: unified-generation baseline; Apache License 2.0.
- **PlanGen** [18]: layout-planning and generation baseline; Apache License 2.0 for the official code.
- **Flow-GRPO** [29]: reinforcement learning implementation; MIT License.
- **T2I-CompBench++** [21]: compositional text-to-image evaluation; MIT License.
- **GenEval**: object-level compositional text-to-image evaluation; MIT License.
- **DPG-Bench**: prompt-following evaluation; Apache License 2.0 for the official ELLA repository.
- **ImgEdit-Bench** [47]: image-editing evaluation; Apache License 2.0, as specified in the official dataset card.
- **RefCOCO** [23]: visual-grounding evaluation; Apache License 2.0 for the official API, with MS COCO images subject to the MS COCO Terms of Use.
- **MMMU** [50]: multimodal-understanding evaluation; Apache License 2.0.
- **POPE** [25]: object-hallucination evaluation; MIT License.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have clearly stated the contributions and scope of the paper in the abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of the work in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: We do not have any theoretical results in this paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have provided all the information needed to reproduce the main experimental results in the appendix and supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have provided our codebase in the supplemental material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided all the training and test details in our paper and supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We have provided statistical significance tests for our main results in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided the compute resources for our experiments in Appendix D.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have followed the NeurIPS Code of Ethics in our paper.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed the potential positive and negative societal impacts of our work in the paper.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We have described the safeguards that have been put in place for responsible release of data or models that have a high risk for misuse in the appendix.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have properly credited the creators or original owners of the assets used in our paper and explicitly mentioned the license and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have provided detailed documentation for the new assets introduced in our paper and the documentation is provided alongside the assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: We do not involve crowdsourcing nor research with human subjects in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: We do not involve research with human subjects in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We have described the usage of LLMs in the appendix.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.