Thinking with Visual Primitives

Ruijie Lu[1,2,*] Yiyang Ma[2,*] Xiaokang Chen[2,*,‡] Lingxiao Luo[2,3,*]
Zhiyu Wu[2,*] Zizheng Pan[2,*] Ting Chen[3] Gang Zeng[1]
[1]State Key Lab of General AI, Peking University [2]DeepSeek-AI [3]Tsinghua University
[*]Core contributors [‡]Project lead

Abstract

Despite the remarkable progress in Multimodal Large Language Models (MLLMs), the prevailing Chain-of-Thought (CoT) paradigms remain predominantly confined to the linguistic space. While recent advancements have addressed the **Perception Gap** through high-resolution cropping (e.g., Thinking with Images), maintaining precise visual references throughout multi-step reasoning remains challenging. We refer to this challenge as the **Reference Gap**: natural-language descriptions can be ambiguous when referring to specific objects, locations, or trajectories in complex visual scenes. In this work, we introduce **Thinking with Visual Primitives**, a novel reasoning framework that elevates spatial markers (*e.g.*, points and bounding boxes) to "minimal units of thought". By interleaving these visual primitives directly into the reasoning process, our model can "point" while it "reasons", anchoring its intermediate reasoning steps to image coordinates. With a relatively low image-token budget, our model achieves frontier-competitive performance on a focused suite of challenging visual QA tasks, matching or exceeding models such as GPT-5.4, Claude-Sonnet-4.6, and Gemini-3-Flash under the evaluated settings. This demonstrates a path toward more efficient and scalable System-2-like multimodal intelligence.

1 Introduction

The convergence of Large Language Models (LLMs) and computer vision has ushered in an era of Multimodal Large Language Models (MLLMs) with sophisticated scene understanding. However, as we push these models toward complex reasoning, often conceptualized as Daniel Kahneman's "System 2" thinking [1], a fundamental limitation of the current paradigm emerges. While the reasoning traces of these models, typically expressed as Chain-of-Thought (CoT), have become increasingly robust in the linguistic domain, they remain largely disjointed from the visual domain.

Specifically, recent efforts to enhance multimodal reasoning mainly focus on the Perception Gap, where the frontier models [2, 3, 4, 5, 6, 7] employ cropping and dynamic tiling on high-resolution images to capture more fine-grained details. However, even with improved fine-grained perception, these models can still make reasoning errors in tasks involving complex spatial layouts or dense object interactions. We identify this failure as the Reference Gap: the inherent inability of natural language to serve as a precise, unambiguous pointer within a continuous visual space. In scenarios like dense counting or multi-step spatial deduction, the model's linguistic "thoughts" lose track of the visual entities they intend to reference, leading to cascading hallucinations. While recent works [8, 9, 10] have explored integrating bounding boxes into the CoT process, they primarily treat grounding as a post-hoc verification mechanism to enhance perception-heavy tasks. These approaches are often confined to high-resolution benchmarks where the challenge is "seeing" rather than "reasoning", and some rely on labor-intensive supervision, which further limits scalability. More importantly, they struggle to address the Reference Gap in complex structural reasoning such as topological navigation, where visual markers must function as explicit reasoning units rather than merely verifiable evidence.

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

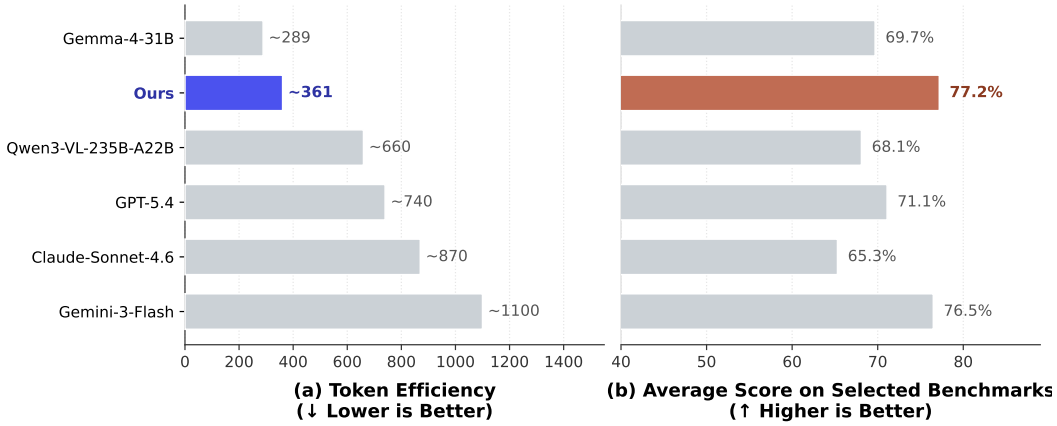


Figure 1: **(a) Token consumption across various models for an image of 800×800 resolution. (b) Average performance across 7 benchmarks, including counting and spatial reasoning, with in-house benchmarks excluded**. We note that the reported scores cover only a subset of evaluation dimensions that are directly relevant to the research focus of this paper, and are therefore not indicative of the models' overall capabilities. For an image of 800×800 resolution, our model retains only approximately 361 image tokens. Best viewed on screen.

In this work, we introduce a new reasoning framework: **Thinking with Visual Primitives**. Rather than treating visual grounding as a secondary task or a final output, we elevate spatial markers (*e.g.*, points and bounding boxes) to "minimal units of thought" that are interleaved directly into the model's reasoning trajectory. This mechanism draws inspiration from human cognitive processes. When navigating a complex maze or counting a dense collection of objects, humans naturally employ deictic pointers (*e.g.*, finger gestures) to reduce cognitive load and maintain logical consistency. By interleaving visual primitives into the thinking process, our model mimics this "point-to-reason" synergy, effectively anchoring abstract linguistic thoughts onto concrete spatial coordinates.

Through extensive benchmarking, we demonstrate that Thinking with Visual Primitives delivers a significant leap in reasoning accuracy. Our model achieves competitive performance, standing on par with or surpassing the latest iterations of GPT, Claude, and Gemini across a wide spectrum of challenging spatial reasoning and visual QA tasks (see Fig. 1). Our findings suggest that the future of multimodal intelligence lies not just in seeing more pixels, but in developing more precise and less ambiguous referential mechanisms that bridge the gap between language and the visual world.

2 Method

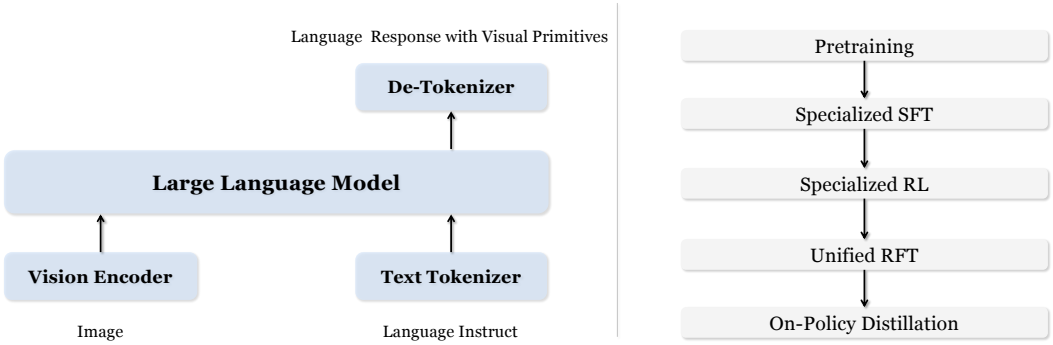
2.1 Overview

This section first introduces the model architecture. Before post-training, the model is endowed with basic grounding and pointing capabilities via broad Internet-scale pretraining; details of the pretraining stage are provided in Appendix. We then elaborate on the post-training pipeline, as illustrated in Fig. 2, and describe the corresponding training data.

2.2 Architecture

Our model adopts a standard architecture similar to LLaVA [11, 12]. Specifically, input images are processed by a Vision Transformer (ViT) to extract visual features, which are then concatenated with language instructions to form an interleaved sequence of vision-language tokens. This sequence is subsequently fed into the Large Language Model (LLM) to generate responses. The language backbone is instantiated with an internal LLM, a Mixture-of-Experts (MoE) model.

For visual encoding, we employ an in-house ViT trained from scratch that supports arbitrary-resolution inputs. It first partitions the input image using a 14×14 patch size to generate patch tokens. Subsequently, at the ViT output, we apply a 3×3 spatial token compression (compressing every 9 adjacent patch tokens into a single token along the channel dimension).



(a) Architecture (b) Training Pipeline

Figure 2: **Model architecture and training pipeline.** Developed upon an internal LLM, our model acquires basic visual primitive generation capabilities during the pretraining phase. This is followed by a post-training stage employing an expert-wise specialization and consolidation paradigm.

2.3 Task Design & Cold-Start Data

Cold-Start Data for Post-Training. While pretraining equips the model with general multimodal priors and basic visual primitive capabilities, post-training (Specialized SFT/RL and the subsequent unified RFT, which denotes a rollout-based fine-tuning stage where SFT data are constructed from verifier-filtered rollouts of an RL checkpoint) needs a small but high-precision cold-start dataset to bootstrap instruction following and reward learning under our visual primitive output interface. Concretely, we construct cold-start data with (i) explicit supervision targets derived from annotations (e.g., boxes/points) or programmatically generated, (ii) automatic verifiers (e.g., rule-based checkers) whenever possible to reduce label noise. We selected representative tasks that benefit from visual primitive-based reasoning (via boxes or points), and designed our cold-start data across four key dimensions: counting, spatial reasoning & general visual QA, maze navigation, and path tracing.

2.3.1 Maze Navigation

While MLLMs have shown proficiency in solving advanced scientific problems, a robust paradigm for topological reasoning remains challenging. Purely linguistic CoT struggles to accurately describe trajectories of irregular shapes. To address this gap, Thinking with Visual Primitives, which could employ points as cognitive units, is uniquely suited for such challenges. We first introduce a maze navigation task that requires the model to determine the solvability of a maze—a process that demands a fundamental understanding of spatial connectivity and reachability. We construct our cold-start data through synthetic data generation, the details of which are provided below.

Design Methodology. We use Depth-First Search (DFS), Prim, and Kruskal algorithms to produce solvable and non-trivial mazes. All three algorithms generate challenging mazes where only few paths exist between any two cells, ensuring solutions that cannot be trivially guessed. We design three maze topologies: rectangular grids, circular mazes composed of concentric rings with angular sectors, and hexagonal (honeycomb) lattices. To enhance model robustness, we additionally designed a series of unsolvable mazes. We first generate a solvable maze and obtain the solution paths, then deliberately place a few walls around the middle of that path—avoiding areas too close to the start or end. This breaks the connectivity in a less obvious way, making the maze appear solvable at first glance, but actually requiring a full search to confirm that no valid path exists. We apply diverse visual styles including gradient and extra-thick walls, varied background patterns, multiple marker types, and random small-angle rotations to prevent overfitting to specific visual patterns.

Difficulty Control. The difficulty of maze navigation largely depends on how many visual reasoning steps the model needs to chain together. We control this by changing the grid size. As the grid becomes larger, the model has to parse more cells, track connectivity over longer distances, and deal with more dead ends that require backtracking. Each of these adds to the overall reasoning complexity. Concretely, easy mazes require the model to chain only a handful of local connectivity checks, while nightmare-level mazes demand sustained, long-range composition of hundreds of such primitive

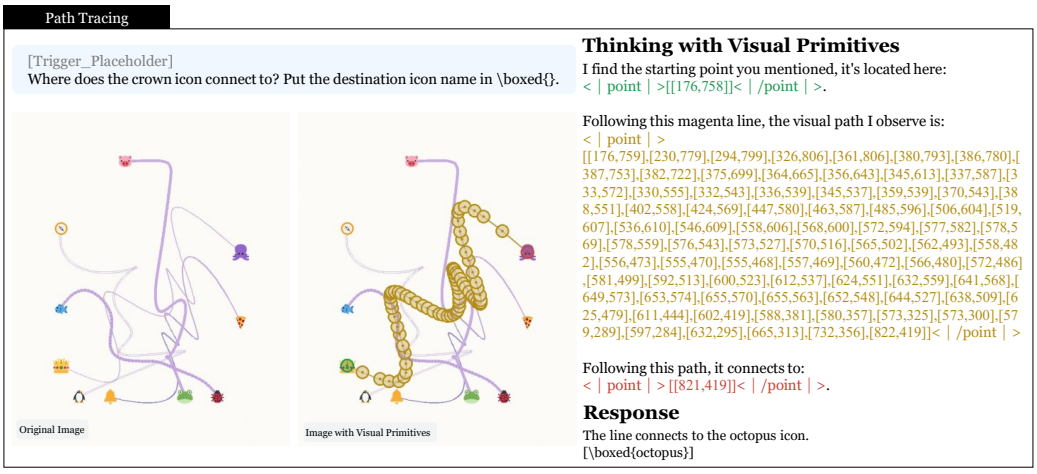


Figure 3: **Example of cold-start data for the path tracing task.** The model identifies the start and end points, then traces the line using visual primitives. Best viewed on screen with zoom.

operations without losing track of previously explored regions. We enforce minimum resolution thresholds at each difficulty level to ensure that the visual primitives remain perceptible, even in the hardest configurations. This ensures that task difficulty stems from reasoning complexity.

Thinking Content Synthesis. We design several natural language formats and templates to produce descriptions of the DFS-based exploration process, including forward exploration and backtracking. Each exploration step is grounded to the image via pointing coordinates, explicitly converting visual primitive operations—checking wall connectivity at a cell, advancing to an adjacent cell, or retreating from a dead end—into verbalized reasoning chains. This serves as the cold start supervision for teaching the model to think with visual primitives rather than merely perceive them. The final output indicates whether the maze is solvable and, if so, provides a verified solution path.

In total, we generate $460,000$ cold-start samples for the task of Maze Navigation.

2.3.2 Path Tracing

In addition to the maze navigation task, we further design a path tracing task to enhance the model's ability to leverage visual primitives for reasoning across diverse scenarios. The task asks the model to follow a specified curve through a tangle of overlapping lines to identify the endpoint it reaches. We instantiate this task as line tracing through procedurally generated images of entangled curves, where each line connects a uniquely labeled start point to an endpoint.

Design Methodology. We generate images which consist of multiple Bézier curves, each connecting a labeled start point to a labeled endpoint. The central challenge lies in intersection disambiguation: wherever two lines cross, the model must invoke a local geometric-continuity primitive to decide which branch continues the target curve. To ensure this primitive is genuinely tested, we carefully prevent any endpoint from overlapping with or being crossed by an unrelated line, discarding and regenerating configurations that violate these constraints. We further include a uniform-style mode in which every line shares the same color and stroke width, stripping away color-based shortcuts and forcing the model to rely solely on curvature continuity at crossings—a direct test of whether the path-tracing primitive has been internalized rather than approximated by color matching. Difficulty scales naturally with the number of lines and their curvature amplitude: easy instances present a few gently curving lines with sparse crossings, while harder instances pack many tightly wound curves into the canvas, multiplying the intersection points at which the figure-ground primitive must be applied. Image resolutions, aspect ratios, and visual styles (color palettes, line styles, endpoint markers, backgrounds) are all randomized to prevent superficial pattern matching.

Thinking Content Synthesis. We explicitly represent the path-tracing process as a sequence of coordinates sampled along the target curve, which reflects how the model attends to and follows the path across the image. The process starts by locating the queried start point, then follows the curve

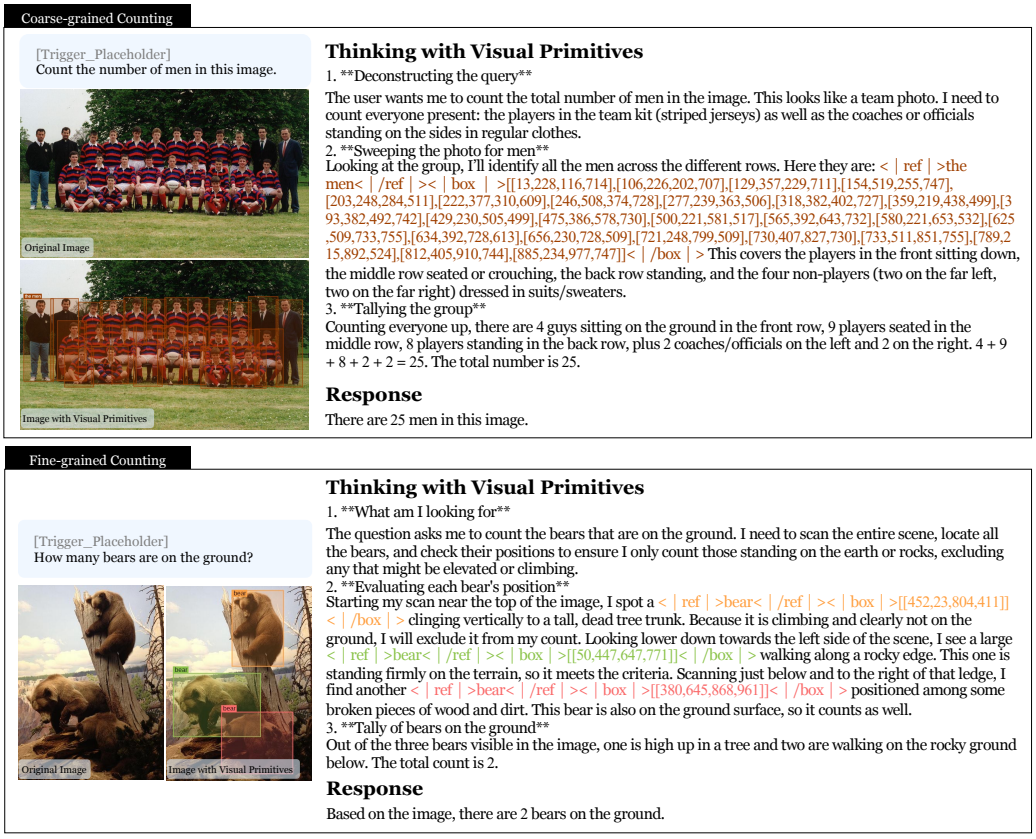


Figure 4: **Illustrative examples of cold-start data for coarse-grained and fine-grained counting**. The model performs intent decomposition and utilizes visual primitives to anchor all pertinent entities, followed by a systematic counting procedure grounded in the visual domain.

through a series of intermediate waypoints, and finally identifies the endpoint reached. Importantly, the density of these waypoints adapts to the local geometry of the curve. Straightforward segments are represented with fewer points, while highly curved regions or dense intersections are described with finer-grained coordinates, mirroring how a human would slow down and pay closer attention in visually complex regions.

In total, we generate $125,000$ cold-start samples across different difficulty levels for the task of Path Tracing. An example is shown in Fig. 3.

2.3.3 Counting

Multimodal Large Language Models consistently struggle with accurate counting, particularly in dense scenes. Unlike humans, who typically employ a systematic scanning-and-accumulation strategy, language-based models often fail to establish precise object correspondences when the object count is high. We address this fundamental bottleneck by employing bounding boxes as visual primitives to provide explicit referential anchors.

Task Decomposition. We categorize counting tasks into two types: *Coarse-grained Counting* and *Fine-grained Counting*. The former focuses on counting general categories (e.g., "dogs"), while the latter requires distinguishing objects based on specific attributes or spatial constraints (e.g., "white dogs" or "the dog on the left").

Coarse-Grained Counting. We aggregate data from multiple dense detection datasets, including [13, 14, 15, 16, 17, 18, 19]. To ensure data quality, we implement a filtering process based on three primary criteria: avoiding excessive object density, ensuring bounding boxes are sufficiently large for clear identification, and maintaining a high recall rate for ground-truth box annotations. For the filtered samples, we prompt an MLLM to generate thinking content and concise final response based on the

images and box annotations. The thinking content generation follows a structured three-step protocol: (1) *Intent Analysis*, where the model identifies the target category; (2) *Batch Grounding*, where the model utilizes visual primitives to locate all candidate objects simultaneously (we found that batch grounding is more efficient for coarse-grained tasks as it leverages the model's inherent localization strengths while preventing repetitive enumeration); and (3) *Statistical Summation*, based on the visual primitives. To eliminate noise during cold-start training, we implement a strict verification mechanism to ensure that all box visual primitives in the thinking content strictly align with the metadata coordinates, follow the predefined syntax, and match the final numerical count.

Fine-Grained Counting. Due to the scarcity of publicly available datasets specifically for fine-grained counting, we developed a specialized data construction pipeline. (1) *Question Generation*: Leveraging the images and scene-graph metadata from GQA [20], we prompt an MLLM to curate informative fine-grained counting questions. Samples that fail to yield meaningful questions are discarded. For each valid sample, we record the ground-truth object IDs, the IDs of excluded negative candidates, and the underlying rationale for the QA pair construction. (2) *Thinking Content Synthesis*: Using the image, scene graph, and the previously generated question (along with its associated IDs and rationale) as input, we guide the MLLM to synthesize a reasoning chain integrated with visual primitives. While the overall thinking structure mirrors that of coarse-grained counting, the model is explicitly instructed to perform a *sequential scan*—systematically identifying and verifying each possible object in the scene against the specified fine-grained constraints. We also applied this methodology to construct negative samples where the ground-truth count is zero, thereby enhancing the model's robustness against hallucinations.

In total, we have approximately $10,000$ cold-start samples for the counting task.

2.3.4 Spatial Reasoning and General Visual QA

We consolidate spatial reasoning and general VQA into a unified category. This integration effectively mitigates the referential ambiguity and semantic drift inherent in purely linguistic descriptions. In constructing our cold-start data, we prioritize spatial reasoning tasks, under the hypothesis that the capability to think with visual primitives developed here will naturally generalize to broader VQA scenarios. Our data curation covers both natural and synthetic environments.

Data Construction in Natural Scenes. Utilizing the images and scene graphs from GQA [20], we prompt an MLLM to design questions centered on spatial relations and object interactions, along with corresponding thinking content. The generated thinking content follows a structured process, including intent analysis, object grounding, and relational inference. To resolve potential ambiguities in crowded scenes, the model is instructed to select distinctive objects and apply multi-attribute constraints (e.g., combining actions and properties) to uniquely specify the target. However, due to the relatively simple relational structure in GQA, it remains challenging to generate complex, multi-hop reasoning samples at scale. To overcome this limitation and fully unlock the model's potential, we further incorporate complex synthetic data.

Data Construction in Synthetic Scenes. We leverage the CLEVR [21] toolchain to generate multi-hop reasoning data. This framework supports controllable scene generation with varying object densities, along with question generation and programmatic execution traces that map each reasoning step to object-level references (e.g., specific object IDs). To supervise the generation of visual primitives, we project 3D object coordinates onto 2D bounding boxes based on the official toolchain. Given the rendered images, scene graphs, questions, answers, and execution traces, we prompt the MLLM to synthesize "Thinking with Visual Primitives" chains, which include intent analysis, task decomposition, and multi-hop grounded reasoning.

Negative Sample Augmentation. To enhance the model's reliability, we construct negative training samples where the queried objects or relationships do not exist. In such cases, the model is trained to provide a "faithful refusal" based on the visual evidence rather than generating fabricated responses.

In total, we generated $9,000$ cold-start samples for the spatial reasoning and general VQA domain.

2.4 Post-Training Pipeline

To maximize the learning efficiency of the model for both box and point visual primitives, our post-training pipeline adopts a "train-specialists-then-merge" strategy, as detailed below. This design accommodates heterogeneous task-specific reward objectives before consolidating the specialists into a unified model.

2.4.1 Specialized SFT

In the Specialized SFT phase, the overall training data consists of 70% general multimodal and pure-text data, and 30% specialized "thinking with visual primitives" data. We conduct SFT separately using the two types of cold-start data constructed in Section 2.3: box (*thinking with grounding*) and point (*thinking with pointing*), preventing mode conflicts. After this training phase, we obtain two specialized models, denoted as $\mathbf{F}_{\mathrm{TwG}}$ and $\mathbf{F}_{\mathrm{TwP}}$.

2.4.2 Specialized RL

Subsequently, we apply Reinforcement Learning (RL) independently to both $\mathbf{F}_{\mathrm{TwG}}$ and $\mathbf{F}_{\mathrm{TwP}}$. Following [22], we utilize the Group Relative Policy Optimization (GRPO) algorithm and follow their hyper-parameters. For counting, spatial reasoning, and general VQA, RL does not require ground-truth annotations for intermediate visual primitives; images, questions, and final answers suffice for constructing the training data. This reduces annotation requirements and enables the use of a broader range of QA data. For synthetic maze navigation and path tracing, we additionally use programmatically available geometric information to compute trajectory-level rewards, providing fine-grained feedback on intermediate reasoning steps without requiring manual trajectory annotations.

During training, we design several Reward Models (RMs) to provide concurrent supervision for each task from three perspectives: format constraints, quality constraints, and accuracy constraints. The first two constraints are shared across different tasks, while the final accuracy constraint requires specific designs tailored to the task type.

Format RM. This RM evaluates the output based on rules to generate a reward score ranging from 0 to 1. Specifically, it verifies whether the representation format of the visual primitives generated by the model is correct. For *thinking with grounding*, this RM additionally checks for redundancy in the model's output, such as generating duplicate bounding boxes; this effectively mitigates the issue of the SFT model falling into an infinite loop of box generation.

Quality RM. This is an LLM-based Generative Reward Model (GRM). It takes the model's thinking content and final response as inputs, and evaluates response quality in terms of redundancy, consistency between thinking and final answer, self-contradictions during the "thinking with visual primitives" process, the meaningfulness of referred box-level entities, and potential "reward hacking" behaviors, such as fabricating a fake ground truth identical to its own prediction to deceive the reward model. Ultimately, the model outputs a score from three discrete tiers $[0.0, 0.5, 1.0]$ and provides a rationale for the given score. Below, we will introduce the Accuracy RM specific to each task.

Accuracy RM for Counting. To provide a smooth and informative learning signal, we design a rule-based counting reward model that captures the degree of deviation between the predicted and ground truth, instead of relying on binary exact-match supervision. Specifically, we apply a smooth exponential decay over the relative error, so that near-correct predictions are only lightly penalized, while larger mistakes receive significantly lower scores. The reward R is given as:

$$R(\hat{y}, y) = \alpha \cdot \exp\left(-\beta \cdot \frac{|\hat{y} - y|}{|y| + 1}\right), \tag{1}$$

where $\hat{y}$ and y denote the predicted and ground-truth counts, respectively. The normalization term $|y| + 1$ makes the reward depend on relative error, allowing small deviations to be more tolerable in scenes with larger object counts. The coefficients α and β control the overall reward scale and decay rate, respectively. In practice, we set $\alpha = 0.7$ and $\beta = 3$, which are empirically chosen.

Accuracy RM for Spatial Reasoning and General VQA. For these tasks, we design an LLM-based GRM. We feed the model's thinking content, its final response, the user query, and the ground-truth answer into the GRM to independently evaluate and score the thinking process and the response. The final reward is calculated as the average of the two scores.

Accuracy RM for Maze Navigation. To encourage reliable maze exploration, we design a rule-based RM whose final reward is a weighted combination of several interpretable components. Specifically, we measure causal exploration progress by sequentially validating the model's exploration trace and truncating it after the first wall violation, then rewarding how close the valid explored region reaches the endpoint. For unsolvable mazes, we instead evaluate exploration completeness by measuring the fraction of reachable cells covered by the valid trace. In addition, we penalize all wall-violating transitions in the full trace, verify the validity of the final path when the model predicts

the maze is solvable, and check whether the final solvability judgment matches the ground truth. This decomposition provides dense reward signals for both intermediate visual reasoning and final answer correctness, rather than relying only on a binary outcome.

Accuracy RM for Path Tracing. To enforce line-following behavior, we design a rule-based RM with a weighted reward over trajectory accuracy, endpoint accuracy, trajectory continuity, and final answer correctness. The trajectory score bidirectionally compares the predicted and ground-truth polylines, penalizing both deviations from the true curve and incomplete coverage of it. We also verify the predicted start and end locations, penalize discontinuous jumps from the traced trajectory to the predicted endpoint, and check whether the final endpoint label matches the ground truth. This provides dense supervision for complete and accurate path tracing beyond the final answer alone.

RL Data. During RL, we expand the data pool and generate N rollouts per sample using the SFT cold-start model ($\mathbf{F}_{\mathrm{TwG}}$ or $\mathbf{F}_{\mathrm{TwP}}$). Samples are grouped by the number of RM-verified correct rollouts k: easy ($k = N$), normal ($1 \leq k < N$), and hard ($k = 0$). We retain only normal-level samples for RL, since they provide informative reward variation for GRPO. After Specialized RL, we obtain two expert models, $\mathbf{E}_{\mathrm{TwG}}$ and $\mathbf{E}_{\mathrm{TwP}}$.

2.4.3 Unified RFT

Equipped with the robust expert models $\mathbf{E}_{\mathrm{TwG}}$ and $\mathbf{E}_{\mathrm{TwP}}$ obtained above, we proceed to integrate the two visual-primitive-based reasoning paradigms—*thinking with grounding* and *thinking with pointing*—into a single unified model. We employ these expert models to perform rollouts over the data pool to generate RFT data. Applying the previously introduced difficulty categorization criteria, we retain all samples classified as "Normal-Level" and randomly sub-sample 5% of the "Easy-Level" data (to prevent catastrophic forgetting in excessively simple scenarios). Leveraging this larger and more diverse RFT dataset, we initialize from the base pretrained model to train an enhanced SFT model. Our RFT training configuration remains identical to that of the SFT cold-start phase (including the training hyper-parameters and the initial checkpoint), with the only difference being the updated training data mixture. Following this procedure, we obtain the unified model $\mathbf{F}$.

2.4.4 On-Policy Distillation

Starting from the unified model $\mathbf{F}$, we further train two stronger specialized teacher models, $\mathbf{E}_{\mathrm{TwG}^+}$ and $\mathbf{E}_{\mathrm{TwP}^+}$, using the same reward and hyperparameters as in the previous RL stage. Although the RFT model $\mathbf{F}$ demonstrates substantial improvements over the cold-start models $\mathbf{F}_{\mathrm{TwG}}$ and $\mathbf{F}_{\mathrm{TwP}}$ in their respective domains, a noticeable performance gap remains when compared to the expert models $\mathbf{E}_{\mathrm{TwG}^+}$ and $\mathbf{E}_{\mathrm{TwP}^+}$. To bridge this gap, we follow [22] and employ On-Policy Distillation (OPD) to effectively consolidate the capabilities of the expert models into a single unified model. This distillation process is achieved by enabling the student model to learn the output distribution of the teacher models based on its own generated trajectories. Formally, given a set of N expert models $\{\pi_{E_1}, \pi_{E_2}, \ldots, \pi_{E_N}\}$, the OPD objective function is defined as:

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \sum_{i=1}^{N} w_i \cdot \mathrm{D}_{\mathrm{KL}}\left(\pi_\theta \parallel \pi_{E_i}\right), \tag{2}$$

where w_i denotes the weight assigned to each expert model, D_{KL} represents the reverse KL divergence loss, and π_θ denotes the student model. We adopt full-vocabulary logit distillation for our OPD implementation. In practice, we utilize two teacher models, including $\mathbf{E}_{\mathrm{TwG}^+}$ and $\mathbf{E}_{\mathrm{TwP}^+}$.

3 Experiments

3.1 Evaluation Setup

Our evaluation framework integrates widely adopted public benchmarks with a curated in-house suite. While public benchmarks are vital for standardized comparisons, their constrained evaluation dimensions often fail to capture the full spectrum of a model's capabilities, e.g., thinking with visual primitives. To bridge this gap, our in-house suite introduces more diverse and challenging axes, serving as a critical complement to the public datasets.

Table 1: **Comparison with frontier models**. To ensure a fair comparison, we evaluated all models via their respective APIs using shared task prompts, with an additional mode-activation prefix for our model. The best results are highlighted in bold; the second-best results are underlined.

Category	Benchmark (Metric)	Gemini-3-Flash	GPT-5.4	Claude-Sonnet-4.6	Gemma4-31B	Qwen3-VL 235B-A22B-Thinking	Ours
Counting	CountQA (EM / RA@10)	**66.1 / 75.1**	48.3 / 60.3	34.8 / 46.6	43.2 / 54.6	42.7 / 54.8	64.9 / 74.1
	Pixmo-Count (EM)	88.2	76.6	68.7	82.9	77.2	**89.2**
	Finegrained_Counting (EM)	79.1	84.2	82.6	79.5	87.2	**88.7**
Spatial Reasoning & General VQA	MIHBench (ACC)	83.2	83.5	81.7	82.2	75.1	**85.3**
	SpatialMQA (ACC)	67.0	61.9	58.2	60.6	54.5	**69.4**
	EmbSpatial (ACC)	82.6	80.9	75.1	82.1	**83.7**	**83.7**
	CV-Bench (ACC)	**88.6**	87.5	85.1	87.5	88.1	88.4
	OmniSpatial (ACC)	**59.6**	58.8	53.2	49.4	55.3	59.5
	Spatial_Reasoning (ACC)	93.2	81.1	97.2	77.2	96.8	**98.7**
Topological Reasoning	Maze_Navigation (ACC)	49.4	50.6	48.9	49.8	49.6	**66.9**
	Path_Tracing (ACC)	41.4	46.5	30.6	33.9	24.5	**56.7**

Public Benchmarks. We evaluate counting with CountQA [23] and Pixmo-Count [24], following their standard protocols and using the official test split for Pixmo-Count. For spatial reasoning and general VQA, we further evaluate on SpatialMQA [25], CV-Bench [26], EmbSpatial [27], OmniSpatial [28], and MIHBench [29].

In-House Benchmarks. To more directly assess "Thinking with Visual Primitives", we curate an in-house benchmark suite covering fine-grained counting, multi-hop spatial reasoning, and topological reasoning. For FINE-GRAINED COUNTING, we generate attribute- and location-conditioned counting queries with hard negatives, followed by manual verification, yielding 600 high-quality test cases. For SPATIAL REASONING, we sample 1,000 true/false and 1,000 open-ended questions from CLEVR [21], converting the latter into multiple-choice questions with MLLM-generated distractors for standardized evaluation. For topological reasoning, we construct MAZE NAVIGATION and PATH TRACING following Sections 2.3.1 and 2.3.2, each containing 2,000 instances.

3.2 Comparison with Frontier Models

For fair comparison, we adopt a unified evaluation protocol across all models. Given that some legacy public benchmarks contain low-resolution images, we apply a preprocessing step to ensure data quality. Specifically, any image with a total pixel count below $640,000$ is upscaled to reach this pixel threshold while strictly preserving its original aspect ratio. For frontier models that support configurable reasoning or thinking budgets (e.g., GPT and Gemini-3-Flash), we uniformly set the thinking budget to low for all evaluations. For all other benchmarks, we follow the official evaluation protocols and metrics. Results are shown in Table 1. Benefiting from the ability to think with visual primitives, our model achieves competitive performance on these tasks with remarkable token efficiency. Notably, all frontier models exhibit suboptimal performance on topological reasoning tasks, suggesting that substantial room for improvement remains in the reasoning capabilities of multimodal large language models.

3.3 Ablation Study

We report the results across different training stages in Table 2. As shown in the table, RFT/OPD closely matches RL Stage I/II on most benchmarks. Taking MIHBench as an example, the score after SFT is 80.2, which significantly improves to 84.0 following the first specialized RL stage. After conducting RFT using the data rolled out from this RL checkpoint, the performance stands at 83.1, trailing the specialized model by 0.9 points. The second RL stage yields a score of 85.7, and the final OPD model achieves 85.3, which is only 0.4 points lower than the specialized model. These results show that RFT and OPD effectively integrate specialized RL gains with minimal degradation from the specialized teachers. Additional ablation studies and qualitative examples are provided in the supplementary material.

Table 2: **Ablation study on training strategies**. We report the results on a subset of benchmarks across different training stages.

Benchmark (Metric)	**SFT (Specialized)**	**RL-I (Specialized)**	**RFT (Unified)**	**RL-II (Specialized)**	**OPD (Unified)**
CountQA (EM / RA@10)	53.9 / 63.9	60.3 / 71.3	61.8 / 71.8	65.4 / 75.0	64.9 / 74.1
MIHBench (ACC)	80.2	84.0	83.1	85.7	85.3
SpatialMQA (ACC)	54.5	64.4	65.6	70.3	69.4
EmbSpatial (ACC)	76.1	81.8	83.2	83.5	83.7
CV-Bench (ACC)	84.8	87.2	87.7	88.7	88.4
OmniSpatial (ACC)	53.6	56.0	57.5	59.6	59.5

3.4 General Multimodal Performance

To assess whether specialized visual-primitive training preserves general multimodal capabilities, we evaluate the pretrained, RFT, and final OPD checkpoints on MMMU [30] and DocVQA [31]. Performance improves from 70.6 to 74.1 on MMMU and from 93.7 to 94.7 on DocVQA after RFT, which includes general-purpose training data. Following OPD, which uses no additional general-purpose data, the scores remain stable at 74.0 and 94.8, respectively. These results indicate that the model retains its general multimodal capabilities on these two benchmarks.

4 Conclusion

To address the inherent "Reference Gap" in Multimodal Large Language Models (MLLMs) during complex reasoning, we introduce "Thinking with Visual Primitives", a novel reasoning framework. Moving beyond the conventional reliance on simply increasing perceptual resolution, we elevate spatial markers (*e.g.*, points and bounding boxes) to "minimal units of thought" and interleave them directly into the model's thinking process. This mechanism endows the model with the ability to "point while it reasons", precisely anchoring abstract linguistic concepts onto physical image coordinates. Our model achieves performance on par with frontier models across highly challenging tasks, including spatial reasoning, visual QA, and topological reasoning, while significantly reducing image token consumption. Our work demonstrates that the path to System-2 multimodal intelligence lies not merely in "seeing more pixels", but in constructing a precise, unambiguous referential bridge between language and vision.

Acknowledgements

This work is supported by the State Key Laboratory of General Artificial Intelligence, China Tower-Peking University Joint Laboratory of Intelligent Society and Space Governance, the National Natural Science Foundation of China (61632003, 61375022, 61403005), and PEK-SenseTime Joint Laboratory of Machine Vision. This work is also supported by grants from the National Natural Science Foundation of China (T2541010), the National Key R&D Program of China (2024YFF1207100, 2024YFF1207103), and Beijing National Research Center for Information Science and Technology (BNRist). The funders had no role in study design, data collection and analysis, publication decisions, or manuscript preparation.

References

[1] Daniel Tumminelli O'Brien. Thinking, fast and slow by daniel kahneman. 2012.

[2] OpenAI. Introducing o3 and o4-mini. `https://openai.com/index/introducing-o3-and-o4-mini/`, 2025.

[3] Jiacong Wang, Zijian Kang, Haochen Wang, Haiyong Jiang, Jiawen Li, Bohong Wu, Ya Wang, Jiao Ran, Xiao Liang, Chao Feng, et al. Vgr: Visual grounded reasoning. *arXiv preprint arXiv:2506.11991*, 2025.

[4] Ji Qi, Ming Ding, Weihan Wang, Yushi Bai, Qingsong Lv, Wenyi Hong, Bin Xu, Lei Hou, Juanzi Li, Yuxiao Dong, et al. Cogcom: A visual language model with chain-of-manipulations reasoning. *arXiv preprint arXiv:2402.04236*, 2024.

[5] Yunze Man, De-An Huang, Guilin Liu, Shiwei Sheng, Shilong Liu, Liang-Yan Gui, Jan Kautz, Yu-Xiong Wang, and Zhiding Yu. Argus: Vision-centric reasoning with grounded chain-of-thought. 2025.

[6] Chaoya Jiang, Yongrui Heng, Wei Ye, Han Yang, Haiyang Xu, Ming Yan, Ji Zhang, Fei Huang, and Shikun Zhang. Vlm-r^3: Region recognition, reasoning, and refinement for enhanced multimodal chain-of-thought. *arXiv preprint arXiv:2505.16192*, 2025.

[7] Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

[8] Haochen Wang, Xiangtai Li, Zilong Huang, Anran Wang, Jiacong Wang, Tao Zhang, Jiani Zheng, Sule Bai, Zijian Kang, Jiashi Feng, et al. Traceable evidence enhanced visual grounded reasoning: Evaluation and methodology. *arXiv preprint arXiv:2507.07999*, 2025.

[9] Yue Fan, Xuehai He, Diji Yang, Kaizhi Zheng, Ching-Chen Kuo, Yuting Zheng, Sravana Jyothi Narayanaraju, Xinze Guan, and Xin Eric Wang. Grit: Teaching mllms to think with images. *arXiv preprint arXiv:2505.15879*, 2025.

[10] Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 37:8612–8642, 2024.

[11] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

[12] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023.

[13] Anindya Mondal, Sauradip Nag, Xiatian Zhu, and Anjan Dutta. Omnicount: Multi-label object counting with semantic-geometric priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19537–19545, 2025.

[14] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H Hsu. Drone-based object counting by spatially regularized regional proposal network. In *Proceedings of the IEEE international conference on computer vision*, pages 4145–4153, 2017.

[15] Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE transactions on pattern analysis and machine intelligence*, 44(11):7380–7399, 2021.

[16] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[17] Mohamed Amgad, Lamees A Atteya, Hagar Hussein, Kareem Hosny Mohammed, Ehab Hafiz, Maha AT Elsebaie, Ahmed M Alhusseiny, Mohamed Atef AlMoslemany, Abdelmagid M Elmatboly, Philip A Pappalardo, et al. Nucls: A scalable crowdsourcing approach and dataset for nucleus classification and segmentation in breast cancer. *GigaScience*, 11:giac037, 2022.

[18] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

[19] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019.

[20] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[21] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.

[22] DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

[23] Jayant Sravan Tamarapalli, Rynaa Grover, Nilay Pande, and Sahiti Yerramilli. Countqa: How well do mllms count in the wild? *arXiv preprint arXiv:2508.06585*, 2025.

[24] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104, 2025.

[25] Jingping Liu, Ziyan Liu, Zhedong Cen, Yan Zhou, Yinan Zou, Weiyan Zhang, Haiyun Jiang, and Tong Ruan. Can multimodal large language models understand spatial relations? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 620–632, 2025.

[26] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Advances in Neural Information Processing Systems*, 37:87310–87356, 2024.

[27] Mengfei Du, Binhao Wu, Zejun Li, Xuan-Jing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 346–355, 2024.

[28] Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. *arXiv preprint arXiv:2506.03135*, 2025.

[29] Jiale Li, Mingrui Wu, Zixiang Jin, Hao Chen, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, and Rongrong Ji. Mihbench: Benchmarking and mitigating multi-image hallucinations in multimodal large language models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 3143–3152, 2025.

[30] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024.

[31] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2199–2208. IEEE, 2021.

[32] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[34] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023.

[35] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

[36] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

[37] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

[38] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[39] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

[40] Jiaqi Wang, Pan Zhang, Tao Chu, Yuhang Cao, Yujie Zhou, Tong Wu, Bin Wang, Conghui He, and Dahua Lin. V3det: Vast vocabulary visual detection dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19844–19854, 2023.

[41] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *ArXiv*, abs/2306.14824, 2023.

Supplementary Material For Thinking with Visual Primitives

A Related Work

Multimodal Large Language Models and Visual Grounding. Recent multimodal large language models (MLLMs) have achieved strong general-purpose visual understanding by connecting vision encoders with large language models and training them on large-scale multimodal corpora or instruction-following data [32, 33, 34, 35, 11, 12]. Beyond generic visual question answering and image-level understanding, a growing body of work further equips MLLMs with explicit grounding abilities, enabling them to associate textual expressions with visual regions, process region-level inputs, or generate spatial coordinates as part of their responses [36, 24]. These studies demonstrate that spatial references such as bounding boxes and points can be represented within the language generation interface. However, in most existing systems, grounding is primarily treated as an output capability or a region-level interaction interface. In contrast, our work studies visual primitives as intermediate units in the reasoning process itself. Rather than merely asking the model to localize objects before or after reasoning, we require it to reason while explicitly maintaining visual references throughout the reasoning trajectory.

Visual Chain-of-Thought and Referential Grounding. Recent works have attempted to extend chain-of-thought reasoning beyond pure language by introducing visual intermediate states into the reasoning process [10, 8, 9]. For example, Visual CoT annotates intermediate bounding boxes and reasoning steps to guide models toward task-relevant regions, while GRIT trains MLLMs to interleave natural language rationales with explicit bounding box coordinates [10, 9]. Another line of work improves multimodal reasoning through active perception, where models crop, zoom, ground, or replay image regions during inference to access fine-grained visual evidence [4, 3, 6, 7]. Visual prompting methods similarly reduce ambiguity by overlaying marks, masks, or boxes on the input image, enabling MLLMs to refer to visual regions more reliably [37]. Despite their effectiveness, these methods primarily address the *perception gap*: they help models see, attend to, or verify relevant visual evidence more faithfully. In contrast, our work focuses on the *reference gap*, where the key challenge is to maintain precise and unambiguous references to visual entities, paths, and spatial states throughout multi-step reasoning.

B Pretraining

B.1 Definition of Visual Primitives

During the pretraining phase, our objective is to equip the model with the fundamental capability to output "visual primitives". We identify two standard output formats in computer vision as primitives: bounding boxes and points. Both representations fulfill the crucial role of spatial referencing. However, they exhibit distinct functional advantages: bounding boxes are adept at capturing the exact location and scale of specific objects, while points are more appropriate for abstract visual referencing, such as tracking motion trajectories or solving topological reasoning problems.

B.2 Motivation for Large-Scale Data Curation

While existing public datasets, such as COCO [38] and Pixmo-Points [24], provide relatively accurate box or point annotations, they suffer from insufficient scale and a notable lack of diversity. To ensure the generalizability of our "Thinking with Visual Primitives" paradigm, it is imperative to curate large-scale web data with rich semantics and high diversity. We prioritize the extensive scaling of bounding box data for the following reasons:

- **Determinism of Annotations**: A bounding box tightly encloses an object, making its annotation relatively deterministic. Conversely, point annotations are highly ambiguous; any coordinate within the object's boundaries can serve as a valid reference, leading to the absence of a strict ground truth. In extreme scenarios involving occlusion, a point intended for a background object might fall onto a foreground occluder, resulting in significant ambiguity.

- **Task Generalizability**: A model trained to output bounding boxes can effortlessly generalize to point-based formats. Since a bounding box can be defined by two points (the top-left and bottom-right coordinates), it inherently encompasses the point representation.
- **Information Richness**: Bounding boxes support a broader range of downstream tasks compared to points. While a point merely provides spatial localization, a bounding box encapsulates detailed geometric information (e.g., width and height). This additional context enables the model to perform more complex reasoning within the "Thinking with Visual Primitives" framework.

B.3 Large-Scale Web Data Construction

Raw Data Acquisition. We acquire a massive volume of internet data related to box grounding by conducting large-scale web scraping across multiple websites. Taking Huggingface as an example, we leverage its official API to download dataset metadata and retain only those with permissive licenses suitable for research, and rigorously exclude all validation and test splits to prevent data leakage during evaluation. To bridge the gap between the heterogeneous column-based storage of these datasets and our unified training format, we employ an LLM-based agent in a cost-effective manner: for each candidate dataset, the agent inspects only the `README.md` and column metadata (field names, data types, and a few sampled values) to determine whether the dataset can be converted into a grounding-compatible structure (e.g., containing bounding box annotations paired with textual labels). It then automatically generates scripts to batch-transform the raw data into a predefined, unified intermediate representation. Across all crawled platforms (including HuggingFace and beyond), following extensive deduplication, we ultimately curate 98K box-grounding-related data sources. Manual inspection of the sampled data reveals a highly diverse range of object categories, spanning from common targets (e.g., people, faces) to domain-specific entities (e.g., lesion regions in CT scans or specific anime characters). However, these raw box annotations still suffer from various issues, such as semantic ambiguity and geometric inaccuracy, necessitating further rigorous filtering. We design a two-step filtering pipeline, as illustrated below.

Step I: Semantic-based Review. Given that the directly crawled datasets are replete with noisy labels unsuitable for vision-language alignment training, we introduce an automated, MLLM-driven semantic review mechanism. While traditional data filtering primarily focuses on the geometric accuracy of bounding boxes, this stage aims to ensure the validity of the semantic label. Specifically, this review process focuses on eliminating three categories of fatal semantic defects:

- **Meaningless Machine Codes and Gibberish**: Many raw datasets retain internal development codes (e.g., pure numeric classes like "0" or "1"). Since these labels lack human-readable natural language semantics, forcing the model to learn such mappings would severely degrade its language generation capabilities. Thus, they are directly discarded.
- **Ungeneralizable Private Entities**: Certain datasets employ specific personal pronouns (e.g., "MyRoommate") or private identifiers (e.g., "ID_Card_1"). Because an MLLM cannot generalize the visual features of a non-public figure from isolated samples (i.e., the visual features of "someone" cannot generalize to a universal concept), such data are strictly filtered out. Conversely, widely recognized celebrities or public figures are retained.
- **Ambiguous Abbreviations and Subjective Evaluations**: Labels common in specific domains (e.g., industrial inspection), such as "OK" or "NG" (Not Good), often lack concrete visual descriptiveness. For instance, a bare "OK" label introduces extreme semantic ambiguity, as an "intact apple" and an "intact circuit board" share absolutely no visual correlation.

For each dataset, we sample three images and prompt the model to calculate a quality score (ranging from 0 to 10) based on the aforementioned criteria. The model then outputs a definitive "KEEP" or "DISCARD" decision, accompanied by a clear justification. This review stage retains 43K out of the initial 98K data sources, which are subsequently advanced to the next filtering phase.

Step II: Visual-Geometric Quality Review. We further evaluate the geometric quality and annotation completeness of the bounding boxes to ensure the model learns precise region-text alignments. This process specifically targets three types of structural annotation defects:

- **Severe Missing Annotations (Low Recall)**: This refers to scenarios where multiple instances corresponding to the given label exist in the image, but only a few are annotated. If a large-scale missing annotation issue (e.g., a miss rate >50%) is detected during sampling, the dataset is immediately discarded.
- **Severe Truncation and Offset**: This occurs when bounding boxes fail to reasonably enclose the target objects. In practice, we adopt a differential tolerance strategy: slightly loose boxes (incorporating minor background noise) are deemed acceptable; however, severe truncations that slice through critical visual features of the object (e.g., cutting off a head or wheels) are strictly unacceptable.
- **Mega Boxes Issue**: If a bounding box meaninglessly covers over 90% of the image area, it is typically indicative of image classification data being forcibly converted into detection data. If this occurs only occasionally within a sampled batch, it is treated as acceptable noise. However, if such global boxes consistently appear across all three sampled images, the dataset is considered to lack meaningful localization information and is discarded.

This review stage further retained 32K out of the 43K remaining data sources. To achieve dataset balance, we design a category-based sampling strategy. For each category within every dataset, we randomly sample N images associated with that class (if the total number of available images for a category is less than N, all are retained). Since a single image may simultaneously belong to multiple categories, we perform global deduplication on the aggregated set after the per-category selection. In practice, we set $N = 1,000$, ultimately yielding over 40 million high-quality samples.

B.4 Unified Pretraining

For general multimodal data, we predominantly utilize large-scale web-crawled data. Regarding the specialized data designed to equip the model with foundational capabilities to output visual primitives, in addition to the aforementioned web crawling and filtering, we also incorporate several high-quality public datasets, such as [39, 38, 40, 24, 41, 19]. We establish a unified formatting standard for both box grounding and point data. For box grounding tasks, we devise several prompt templates, such as "*Locate `TARGET` in this image and report its bounding box coordinates.*", where `TARGET` serves as a placeholder for the queried object. The corresponding response format is formulated as follows: `<|ref|>TARGET<|/ref|><|box|>[[x1,y1,x2,y2],[x3,y3,x4,y4]...]<|/box|>`[1], where `<|ref|>`, `<|/ref|>`, `<|box|>`, and `<|/box|>` are special tokens within the vocabulary. `x1,y1` and `x2,y2` denote the top-left and bottom-right coordinates of the bounding box. These coordinates are normalized to discrete integers ranging from 0 to 999. In scenarios with multiple instances, the bounding boxes are ordered from left to right. Similarly, for point tasks, we design prompt templates such as "*Help me find `TARGET`. Give me the center point for each instance.*" The expected response format is defined as: `<|point|>[[x1,y1],[x2,y2]...]<|/point|>`, where `<|point|>` and `<|/point|>` are special tokens, and `x1,y1` denotes the point coordinates. Notably, in contrast to the box grounding format, the response paradigm for point tasks does not require outputting the object name. This design choice aims to extend point-based representations to more abstract concepts, such as utilizing a sequence of points to denote a trajectory. Ultimately, the whole pretraining phase consumes trillions of multimodal tokens.

C Examples of Cold-Start Data

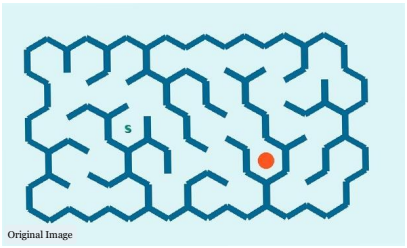
We present an example of our cold-start data for the maze navigation task in Fig. S1. The sample illustrates how the model identifies the start and end points and then explores possible routes in a DFS-style manner, with each reasoning step grounded by visual primitives.

D Additional Ablation Studies

Training-side ablation We train a matched linguistic-CoT baseline using the same pretrained initialization, underlying training samples, RFT recipe, and hyperparameters. We remove bounding-box coordinates from the training reasoning traces while preserving their linguistic structure. As

[1] Please note that all numbers are represented as strings and are encoded into a single token by the tokenizer.

Maze Navigation



Original Image

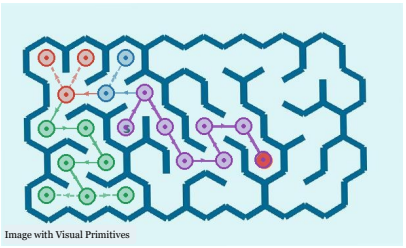


Image with Visual Primitives

[Trigger_Placeholder] Is there a feasible way to get from the lime text label to the tangerine circle? Please draw the route if any. Display \boxed{True} at the end if there is a path, else display \boxed{False}.

Thinking with Visual Primitives

I'll use a trial-and-error strategy to explore this maze. First locate the starting point: < | point | >[[309,512]]< | /point | >, and the destination: < | point | >[[652,644]]< | /point | >.

Start Exploring:

Step1: Reaching < | point | >[[357,369]]< | /point | > after going upper-right, I face 2 directions to choose from (lower-right, left).lower-right direction might also be correct, but I'll explore left direction first.
Step2: Moving left: < | point | >[[357,369],[260,372]]< | /point | >...reaching < | point | >[[260,372]]< | /point | >, I see 2 forks here. Observing that upper-right direction is a straight dead end at < | point | >[[310,226]]< | /point | >, can see wall directly. Still need to explore left direction.
Step3: Walking left: < | point | >[[260,372],[161,375]]< | /point | >...arriving at < | point | >[[161,375]]< | /point | >, this is another junction with 3 paths. Observing that upper-right direction is a straight dead end at < | point | >[[211,226]]< | /point | >, can see wall directly; upper-left direction to < | point | >[[113,226]]< | /point | > is obviously a dead end. Still need to explore lower-left direction.
Step4: Moving lower-left: < | point | >[[161,375],[113,518],[211,515],[261,656],[162,658],[213,796]]< | /point | >...reaching < | point | >[[213,796]] < | /point | >, I see 2 forks here. Observing that right direction is a straight dead end at < | point | >[[310,791]]< | /point | >, can see wall directly; left direction is a straight dead end at < | point | >[[113,791]]< | /point | >, can see wall directly. All directions are obviously dead ends, need to backtrack.
Step5: This path is blocked, backtracking to the previous junction < | point | >[[260,372]]< | /point | >, trying another direction.
Step6: All paths in this branch are explored, all dead ends. Going back to main junction < | point | >[[357,369]]< | /point | >, taking lower-right direction.
Step7: Reaching < | point | >[[357,369]]< | /point | > after going upper-right, I face 2 directions to choose from (lower-right, left).left direction was already explored, it's a dead end.
Step8: Heading towards lower-right: < | point | >[[357,369],[408,510],[457,650]]< | /point | >
Step9: The path is clear, going directly to the exit: < | point | >[[457,650],[554,648],[506,507],[603,507],[652,645]]< | /point | >
Step10: Up ahead < | point | >[[652,645]]< | /point | > is the destination! Through systematic exploration, I find the correct path.

Final Path: After exploration, the correct route is:
< | point | >[[310,513],[357,369],[408,510],[457,650],[554,648],[506,507],[603,507],[652,645]]< | /point | >

Successfully reaching the destination: < | point | >[[652,644]] < | /point | >!

Response

The maze is solvable. The verified path is:
< | point | >[[310,513],[357,369],[408,510],[457,650],[554,648],[506,507],[603,507],[652,645]]< | /point | >
[\boxed{True}]

Figure S1: **Example of cold-start data for the maze navigation task.** The model first identifies the start and end points, then explores possible paths in a DFS manner. Best viewed on screen with zoom.

Table S1: **Ablations of interleaved visual primitives.** The training-side comparison uses matched RFT conditions. The inference-side comparison uses the same final OPD checkpoint with different reasoning formats. A dash indicates an experiment that was not conducted.

	Training-side (RFT)		Inference-side (OPD)	
Benchmark (Metric)	**Linguistic CoT**	**Interleaved primitives**	**Linguistic CoT**	**Interleaved primitives**
CountQA (EM / RA@10)	48.3/61.8	61.8/71.8	48.6/60.0	64.9/74.1
Pixmo-Count (EM)	80.0	88.2	78.5	89.2
SpatialMQA (ACC)	61.5	65.6	65.8	69.4
Path Tracing (ACC)	—	—	30.5	56.7

shown in Table S1, interleaved visual primitives consistently improve performance across the three evaluated benchmarks.

Inference-side ablation We evaluate the same final OPD checkpoint using either purely linguistic CoT or interleaved visual primitives. Switching to linguistic CoT reduces performance across all evaluated tasks. These results support the utility of the interleaved reasoning format at inference time.

E Qualitative Examples

Boxes as Visual Primitives. As shown in Figs. S2 to S4, our model demonstrates strong performance on coarse-grained and fine-grained counting tasks through thinking with grounding, while also

exhibiting emergent capability synergies. For instance, the model is able to integrate world knowledge for visual question answering, perform counterfactual reasoning, and provide actionable suggestions with spatial coordinates tailored to users' everyday needs. Although our post-training data about visual primitives does not include any Chinese corpus, the model is capable of thinking and responding in Chinese, benefiting from the multilingual capabilities inherited from the base model.

Points as Visual Primitives. As shown in Fig. S5, our model exhibits simple topological reasoning capabilities. Specifically, in the maze navigation task, it explores potential paths to ultimately reach the goal; in the path tracing scenario, it leverages visual primitives to track the designated line.

F Limitations

Despite these promising results, our current work has certain limitations. First, constrained by input resolution, the model's performance in fine-grained scenarios remains sub-optimal, leading to occasionally imprecise outputs of visual primitives. This could potentially be addressed by integrating our framework with existing methods targeting the "Perception Gap" to achieve complementary benefits. Second, the current "thinking with visual primitives" capability relies on explicit trigger words for activation. In the future, we aim to enable the model to autonomously determine whether to invoke this mechanism based on the specific context. Third, utilizing points as visual primitives to solve complex topological reasoning problems remains a formidable challenge, and our current model exhibits limited cross-scenario generalization. Exploring ways to broaden the applicability and robustness of this technique constitutes an important direction for future research.

G Negative societal impacts

While Thinking with Visual Primitives improves fine-grained visual grounding, counting, and spatial reasoning, these capabilities could also be misused in privacy-sensitive settings, such as surveillance, tracking, or detailed analysis of people in images and videos. Moreover, if deployed in high-stakes applications, incorrect visual reasoning may lead to harmful decisions. We therefore emphasize that the method should be used with appropriate privacy safeguards, human oversight, and restrictions against surveillance or other harmful uses.

[Trigger_Placeholder]
How many pokemons are there in this photo?



Thinking with Visual Primitives

1. **Understanding the request**
The user wants me to count the number of Pokémon figures in the image. I need to look for characters from the Pokémon franchise among the various toys displayed on the table.
2. **Scanning the table for Pokémon**
Looking at the collection of figures, I can identify the Pokémon characters: < | ref | >pokemons < | /ref | >< | box | >[[215,483,368,711],[330,180,422, 348],[465,438,582,638],[581,403,674,588],[635,195,723,323],[640,244, 814,429]]< | /box | >. Breaking this down visually, I see Pikachu on the left, Psyduck in the back row, Charmander in the center, Bulbasaur in the front row, Jigglypuff in the back right, and Squirtle next to Jigglypuff, partially obscured by the Detective Pikachu figure.
3. **Distinguishing Pokémon from other figures**
I am making sure not to include characters from other franchises. Specifically, I'm ignoring Doraemon on the left, Luffy, Zoro, and the red-hatted figure from One Piece in the front left, the Crayon Shin-chan figure in the back center, the female figure, the guy in the detective hat, and the boxy character on the far right.
4. **Final count of Pokémon**
Counting the distinct Pokémon characters found, there are 6 of them.

Response

Based on the image, there are 6 Pokémon figures.

[Trigger_Placeholder]
Which one is heavier?



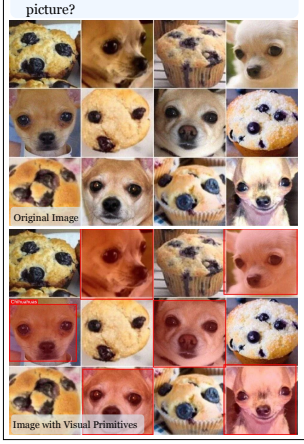
Thinking with Visual Primitives

1. **Understanding the weight comparison task**
The user is asking to determine which of the two objects on the scale is heavier. To answer this, I need to locate each object and evaluate the physical evidence provided by the scale's state (which side is lower and which is higher).
2. **Locating the left object**
On the left side of the scale, I see a < | ref | >large gray metal locker< | /ref | >< | box | >[[26,41,396,699]]< | /box | >. It is constructed with solid, rigid materials and mounted securely on the left < | ref | >metal tray< | /ref | >< | box | >[[20,685,393,759]]< | /box | > of the balance. The locker is rectangular and imposing, suggesting substantial structural mass.
3. **Locating the right object**
On the right side of the scale, resting on the corresponding < | ref | >metal tray< | /ref | >< | box | >[[636,747,970,860]]< | /box | >, is a < | ref | >translucent rainbow bear< | /ref | >< | box | >[[691,407,916,808]]< | /box | >. It looks like a large, vibrant gummy bear made of a translucent, jelly-like substance.
4. **Evaluating the scale's equilibrium**
The visual evidence from the balance scale is very clear. The < | ref | >metal tray< | /ref | >< | box | >[[636,747,970,860]]< | /box | > on the right is positioned lower to the ground than the < | ref | >metal tray< | /ref | >< | box | >[[20,685,393,759]]< | /box | > on the left. The arm holding the right tray is also pointing downward. This means the right side of the scale is heavier.
5. **Conclusion**
Because the right side containing the gummy bear is lower than the left side containing the locker, the bear must be heavier in this scenario.

Response

Based on the scale's tilt, the rainbow bear is heavier.

[Trigger_Placeholder]
How many Chihuahuas are there in the picture?



Thinking with Visual Primitives

1. **Defining the target animal**
The question asks me to count the number of Chihuahuas in the image. I need to focus on differentiating the dogs from the blueberry muffins that also appear in the grid. Looking at the overall layout, the grid seems to contain alternating photos of dogs and muffins.
2. **Scanning the grid for dogs**
I'll search the entire grid to locate every dog. After scanning the rows, I spot the following set of distinct Chihuahuas: < | ref | >Chihuahuas< | /ref | >< | box | >[[2,366,233,641],[247,0,498,341],[255,680,498,997],[499,347,749,670],[743,0,996,317],[743,658,992,997]]< | /box | >. These dogs appear in the first row (second and fourth columns), the second row (first and third columns), and the third row (second and fourth columns). They are mostly a light tan or brownish-white color, with large dark eyes and large ears.
3. **Confirming the final count**
Looking at the identified dogs, I have counted 6 Chihuahuas in total.

Response

There are 6 Chihuahuas in the picture.

Figure S2: **Showcases of thinking with grounding.** Examples include fine-grained counting and counter-commonsense visual question answering. Best viewed on screen with zoom.

[Trigger_Placeholder]
这附近有NBA球队吗?



Thinking with Visual Primitives

1. **分析用户需求**
用户的问题是关于地理位置的：“这附近有NBA球队吗？”。要回答这个问题，我需要先确定图片展示的具体地理位置，然后分析该地区是否拥有NBA球队。
2. **识别图中的世界地标**
我首先观察图片，寻找能够定位的地标。最显著的特征是这座横跨深蓝色水面、鲜艳红色的悬索桥。背景中还可以看到城市天际线和丘陵。通过视觉识别，我确认这是 <|ref|>彩色的金门大桥<|/ref|><|box|>[[0,134,882,795]]<|/box|>，是旧金山（San Francisco）的标志性建筑。
3. **关联地理位置与NBA球队**
既然确认了地点是旧金山，接下来我将城市与NBA联系。旧金山市有一支NBA球队：金州勇士队（Golden State Warriors）。虽然他们曾经在奥克兰比赛，但现在的球馆大通中心（Chase Center）就位于旧金山。
4. **给出结论**
基于位置和球队的存在，这附近确实有NBA球队。

Response

是的，这附近有NBA球队。图中展示的是旧金山的金门大桥，而旧金山正是NBA球队金州勇士队（Golden State Warriors）的主场所在地。

[Trigger_Placeholder]
观察桌面上的设备和原料，我应该如何制作一杯美味的拿铁呢!



Thinking with Visual Primitives

1. **了解目标**
用户希望制作一杯美味的拿铁。我可以在桌面上看到一台全自动咖啡机和相关的制作原料。我需要根据这些设备配置，来确定制作拿铁的具体步骤。
2. **定位咖啡机与原料**
首先，我注意到图中央的主体是一台 <|ref|>黑色自动意式咖啡机<|/ref|><|box|>[[111,107,721,970]]<|/box|>。在它的前端，我看到 <|ref|>蒸汽棒<|/ref|><|box|>[[164,405,236,693]]<|/box|>，这通常用于加热和打发牛奶。右侧放置着一个 <|ref|>不锈钢牛奶壶<|/ref|><|box|>[[670,638,853,905]]<|/box|>，这是制作奶泡的核心工具。旁边还有一袋 <|ref|>咖啡豆包装袋<|/ref|><|box|>[[810,438,995, 885]]<|/box|>，里面装有整颗的 <|ref|>咖啡豆<|/ref|><|box|>[[385,108,603,150]]<|/box|>。
3. **选择饮品程序**
仔细查看咖啡机的操作面板，上面有多个饮品选项。为了制作一杯拿铁，我需要选择代表拿铁的图标并点击它。因此，我点击 <|ref|>拿铁按钮<|/ref|><|box|>[[408,219,444,261]]<|/box|>。
4. **准备奶泡**
选择程序后，下一步是利用 <|ref|>蒸汽棒<|/ref|><|box|>[[164,405,236,693]]<|/box|>来处理牛奶。我拿起旁边的 <|ref|>不锈钢牛奶壶<|/ref|><|box|>[[670,638,853,905]]<|/box|>，将其放置在蒸汽棒下方。接着，操作咖啡机，利用蒸汽棒将壶中的牛奶打发成细腻的奶泡。
5. **准备咖啡杯**
当奶泡准备好后，我只需将其倒入一个合适的杯子中。在咖啡机右侧，我发现了一个空置的 <|ref|>陶瓷咖啡杯<|/ref|><|box|>[[535,779,770,988]]<|/box|>，倒奶泡时可以将其拿过来。同时，通过咖啡机自动将热咖啡液注入杯中，与奶泡结合，最终完成美味的拿铁制作。

Response

要制作一杯美味的拿铁，你可以按照以下步骤使用这台自动咖啡机：
1. 在咖啡机控制面板上点击选择“拿铁”选项。
2. 使用右侧的蒸汽棒将不锈钢奶壶中的牛奶打发成细腻的奶泡。
3. 将打发好的牛奶和奶泡与机器制作的浓缩咖啡混合，倒入左侧准备好的陶瓷咖啡杯中即可。

[Trigger_Placeholder]
这是哪儿?



Thinking with Visual Primitives

1. **初识小巷**
这是一张充满古朴与岁月感的照片，展现了一条幽静的巷弄。整个场景有一种令人放松的安静氛围，似乎在某个历史悠久的古镇或传统村落里。
2. **观察建筑与街道**
目光首先被左侧的建筑吸引，那里有高挑的 <|ref|>深色木制门框<|/ref|><|box|>[[2,249,176,607]]<|/box|>，做工精致，上方还悬挂着 <|ref|>传统的方形灯笼<|/ref|><|box|>[[56,219,118,284]]<|/box|>，增添了一抹古色古香的韵味。木门旁边，一面 <|ref|>泥土质感的墙壁<|/ref|><|box|>[[123,254,341,609]]<|/box|> 裸露着，看起来经过岁月的洗礼而显得斑驳，上面还贴着一些颜色鲜艳的 <|ref|>小型广告海报<|/ref|><|box|>[[182,412,259,510]]<|/box|>，给旧墙带入了些许现代的生活气息。 视线转向街道中央，是一条经典的 <|ref|>鹅卵石铺成的步道<|/ref|><|box|>[[0,554,675,995]]<|/box|>。步道由不规则的石块拼嵌而成，中间的平整石板和两侧圆润的砾石相互交替，纹理非常清晰。 迈步向巷子深处看去，可以看到更远处的 <|ref|>经典老式建筑<|/ref|><|box|>[[320,330,515,555]]<|/box|>，它们保持着传统的坡屋顶和木质结构，层层叠叠，延伸向远方。
3. **探索右侧庭院**
在巷子的右侧，我发现了一扇 <|ref|>带有竖条栅栏的木门<|/ref|><|box|>[[481,433,536,563]]<|/box|>，透着几分神秘。紧挨着这道木门，是一面 <|ref|>老旧的土黄色石墙<|/ref|><|box|>[[525,231,981,726]]<|/box|>，墙面粗糙且颜色饱满。木门上嵌着一块 <|ref|>木质牌匾<|/ref|><|box|>[[556,386,616,472]]<|/box|>，上面刻着字。墙根下的条形石台上，错落有致地摆放着几盆植物和扎染布艺装饰，显得生机盎然。
4. **感受整体氛围**
这条小巷完整地保留了传统建筑风貌与原始的石板路面，加上两侧的植物点缀，呈现出一种耐人寻味的古韵与宁静感。从照片中的老建筑风格、尤其是木门与石墙的构建方式来看，这里极有可能是我曾见过一些资料的中国云南古镇，例如丽江、沙溪或类似的白族、纳西族聚居区。

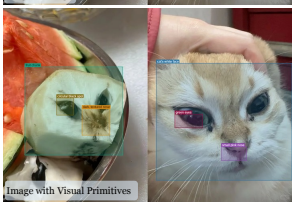
Response

从图片中的木构建筑、泥石墙体、鹅卵石街道和传统的扎染布艺元素来看，这应该是中国云南的一个古镇（例如丽江古城、沙溪古镇或类似的拥有白族/纳西族传统建筑风貌的地方）。

Figure S3: **Showcases of thinking with grounding.** Examples include world-knowledge-based question answering and seeking actionable suggestions. Best viewed on screen with zoom.

[Trigger_Placeholder]
Why is this funny?





Thinking with Visual Primitives

1. **Initial impression of the image**
This is a side-by-side comparison image. The left side shows a chunk of fruit (maybe a pear or apple) with a face-like indentation and dark marks. The right side shows a close-up of a cat's face. It looks like someone carved the fruit to look exactly like the cat.
2. **Locating and explaining the resemblance**
I'm looking for the visual similarities. On the left, I see a <|ref|>fruit chunk<|/ref|><|box|>[[75,303,414,758]]<|/box|>. Within its pale flesh, there are natural dark spots that look like eyes. Specifically, there is a <|ref|>circular black spot<|/ref|><|box|>[[185,463,244,552]]<|/box|> that perfectly mimics a pupil. Additionally, twisted fiber or skin at the bottom of the fruit chunk looks like a <|ref|>dark, textured nose<|/ref|><|box|>[[273,519,369,655]]<|/box|>. Now looking at the right, there is a <|ref|>cat's white face<|/ref|><|box|>[[530,284,999,888]]<|/box|> filling the frame. I can see a pair of <|ref|>green eyes<|/ref|><|box|>[[594,547,690,617]]<|/box|>, looking absolutely devastated or sad. Below the eyes is a <|ref|>small pink nose<|/ref|><|box|>[[755,715,842,786]]<|/box|> located between the cheeks.
3. **Drawing the conclusion**
The humor comes from the 'devastated' expression on the cat's face being perfectly replicated by the randomly shaped marks on the fruit. It looks like the fruit is a raw, unedited version of the cat's soul-piercing sadness. The comparison is spot on, making it funny.

Response

This is funny because the piece of fruit on the left has natural dark spots that form a face looking remarkably similar to the sad, green-eyed cat on the right. The similarities in the "eyes" and "nose" create a hilarious, uncanny resemblance between the two.

[Trigger_Placeholder]
该如何逃出生天呢?





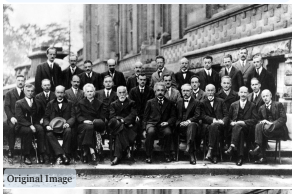
Thinking with Visual Primitives

1. **初步观察**
我环顾四周，发现自己身处一个封闭的室内环境。墙面粗糙，左侧有一扇带有铁栅栏的窗户，右侧是一扇关闭的木门。眼前最引人注目的是高处架子上的一把金色钥匙。我需要找到逃出去的方法。
2. **寻找潜在线索**
我注意到墙面上高悬着<|ref|>一把发光的金色钥匙<|/ref|><|box|>[[566,110,649,165]]<|/box|>，它似乎在暗示我这是解锁的关键。木架上并没有可视的楼梯或踏板。木门下方，我看到<|ref|>一个带有金属门把手和锁孔的深色木制门<|/ref|><|box|>[[665,231,991,910]]<|/box|>，这显然是需要钥匙才能打开的出口。在房间中央的地板上，摆放在我面前的<|ref|>一把简单的木制椅子<|/ref|><|box|>[[399,554,603,936]]<|/box|>。它的位置似乎有些特殊。
3. **整合信息与推理**
目前我面临的核心问题是如何拿到高处架子上的钥匙。我只有一把普通高度的木制椅子，直接伸手是够不到钥匙的。观察门把手和椅子的位置，它们都在房间的较低高度，而光线和我的注意力都集中在上方。我意识到，这很可能是一个简单的物理谜题。如果我移动这把椅子，把它推到门边或架子下方，我或许能够踩在上面够到钥匙。
4. **得出结论**
综合来看，要逃出这个房间，最直接的策略是：先移动这把低矮的椅子，将其挪到悬挂着金色钥匙的墙壁或架子下方，然后利用椅子作为垫脚石，伸手取下钥匙，最后走到右侧的木门处用钥匙开门。

Response

要逃出生天，你可以把木椅移动到挂着钥匙的架子下方，踩着椅子拿到钥匙，然后去开右侧的门。

[Trigger_Placeholder]
How many people are there?





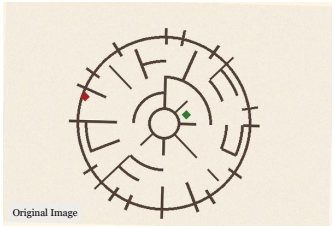
Thinking with Visual Primitives

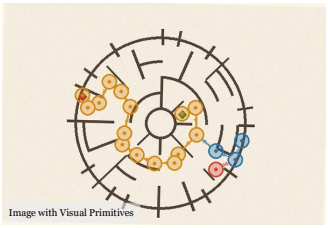
1. **Understanding the task**
I need to count the number of people visible in the image. The picture is a large group portrait featuring several rows of individuals, so I need to scan carefully from the back to the front to ensure no one is missed.
2. **Scanning the group for individuals**
I'll look across the entire frame, identifying heads, faces, and bodies to pick out each individual. Sweeping across the arrangement of people, I find all of them: <|ref|>people<|/ref|><|box|>[[4,459,80,859],[32,506,147,900],[105,477,189,883],[107,335,202,638],[136,502,250,893],[202,362,276,616],[210,463,289,826],[248,499,363,894],[265,386,335,617],[307,455,419,835],[341,383,416,596],[355,510,478,898],[413,392,501,611],[425,447,529,835],[483,489,608,895],[487,374,584,576],[539,450,623,824],[582,374,664,594],[582,498,685,889],[626,471,714,817],[658,371,750,591],[684,508,774,887],[720,469,812,844],[737,365,816,603],[768,510,879,895],[788,320,882,601],[824,470,919,889],[828,370,930,625],[856,529,989,891]]<|/box|>. This includes people in the very back standing tall, those standing in the middle slightly hunched forward to be seen, and the individuals seated on chairs in the very front.
3. **Final count confirmation**
Counting the distinct individuals I just located across the entire group, I find 29 people in total.

Response

There are 29 people in the image.

Figure S4: **Showcases of thinking with grounding.** Examples include humor comprehension in images, escape room game guidance, and counting. Best viewed on screen with zoom.





[Trigger_Placeholder]
Is it possible to navigate from the green rhombus through the maze to the red label? If so, explain the path. Finally, output \boxed{True} or \boxed{False} based on solvability.

Thinking with Visual Primitives

Okay, let me analyze the structure of this maze step by step. First locate the starting point: < | point | >[[555,499]]< | /point | >, and the destination: < | point | >[[247,416]]< | /point | >.

Start Exploring:

Step1: Arriving at < | point | >[[555,499]]< | /point | >, I see 2 paths ahead (counter-clockwise, outward). The counter-clockwise direction is also passable, but from the maze layout, outward direction's path looks wider.
Step2: At < | point | >[[597,472]]< | /point | >, I notice 2 directions to choose from (clockwise, counter-clockwise).Although counter-clockwise direction is also walkable, after observing path structure, I judge clockwise direction more likely leads to exit.
Step3: Reaching < | point | >[[600,594]]< | /point | >, I face 2 forks (clockwise, outward).Let me try outward direction first.
Step4: Moving outward: < | point | >[[600,594],[660,667],[720,705]]< | /point | >...reaching < | point | >[[720,705]]< | /point | >, I see 2 forks here. Observing that up direction is a straight dead end at < | point | >[[740,620]]< | /point | >, can see wall directly. Still need to explore the continuing outward direction.
Step5: Pushing clockwise: < | point | >[[720,705],[660,750]]< | /point | >...but hit a wall at < | point | >[660,750]]< | /point | >. It's a dead end, must backtrack to the previous fork < | point | >[[720,705]]< | /point | >.
Step6: All paths in this branch are explored, all dead ends. Going back to main junction < | point | >[[600,594]]< | /point | >, taking clockwise direction.
Step7: Reaching < | point | >[[544,682]]< | /point | >, I face 3 forks (inward, outward, outward).Comparing to inward direction, outward direction's passage characteristics are more like a main path.
Step8: Reaching < | point | >[[532,720]]< | /point | >, I face 2 forks (clockwise, outward).The outward direction is also passable, but from the maze layout, clockwise direction's path looks wider.
Step9: At < | point | >[[471,722]]< | /point | >, I notice 2 directions to choose from (clockwise, inward).Although inward direction is also walkable, after observing path structure, I judge clockwise direction more likely leads to exit.
Step10: Reaching < | point | >[[415,683]]< | /point | >, I face 2 forks (clockwise, outward).Although outward direction is also walkable, after observing path structure, I judge clockwise direction more likely leads to exit.
Step11: Reaching < | point | >[[371,637]]< | /point | >, I face 2 forks (inward, outward).Comparing to outward direction, inward direction's passage characteristics are more like a main path.
Step12: At < | point | >[[378,567]]< | /point | >, I notice 3 directions to choose from (clockwise, inward, outward).The inward, outward direction is also passable, but from the maze layout, clockwise direction's path looks wider.
Step13: At < | point | >[[396,467]]< | /point | >, I notice 2 directions to choose from (outward, outward).After eliminating dead ends, outward is the best choice.
Step14: Reaching < | point | >[[368,397]]< | /point | >, I face 2 forks (clockwise, outward).The outward direction is also passable, but from the maze layout, clockwise direction's path looks wider.
Step15: At < | point | >[[330,355]]< | /point | >, I notice 2 directions to choose from (outward, outward).After eliminating dead ends, clockwise is the best choice.
Step16: Continuing outward: < | point | >[[330,355],[300,450]]< | /point | >. Keeping this direction.
Step17: Looking in outward direction, I can directly see the path to the exit, no need to linger at the junction: < | point | >[[300,450],[265,472],[247,416]]< | /point | >
Step18: Arriving at < | point | >[[247,416]]< | /point | >, I finally see the exit!

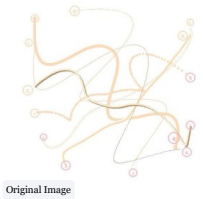
Final Path: After exploration, the correct route is:
< | point | >[[555,499],[597,472],[600,594],[544,682],[532,720],[471,722],[415,683],[371,637],[378,587],[396,467],[368,397],[330,355],[300,450],[265,472],[247,416]]< | /point | >

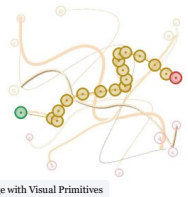
Successfully reaching the destination: < | point | >[[247,416]]< | /point | >!

Response

The maze is solvable. The verified path is:
< | point | >[[555,499],[597,472],[600,594],[544,682],[532,720],[471,722],[415,683],[371,637],[378,587],[396,467],[368,397],[330,355],[300,450],[265,472],[247,416]]< | /point | >
[\boxed{True}]

[Trigger_Placeholder]
What number does C connect to? Show the path and output \boxed{endpoint number}.





Thinking with Visual Primitives

I find the starting point you mentioned, it's located here:
< | point | >[[195,594]]< | /point | >.
Following this beige line, the visual path I observe is:< | point | >[[195,594],[319,617],[338,626],[342,588],[375,547],[438,522],[508,512],[565,505],[604,490],[609,464],[595,427],[579,391],[580,364],[606,358],[661,372],[720,408],[765,442],[803,458]]< | /point | >
Following this path, it connects to: < | point | >[[803,457]]< | /point | >

Response

Based on the visual trace, the endpoint is 5. [\boxed{5}]

Figure S5: **Showcases of thinking with pointing.** Examples include maze navigation and path tracing. Best viewed on screen with zoom.

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: Our abstract and introduction accurately reflect the paper's contributions and scope.

   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 discuss the limitations of our work.

   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: The paper does not include theoretical results.

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: The paper provides detailed evaluation protocols, metrics, model architecture, training pipeline, and implementation details to facilitate reproduction and verification of the main results. However, exact end-to-end reproduction may be limited because some internal resources cannot be fully released due to confidentiality and licensing constraints.

   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: [No]

Justification: We have publicly released a subsequent version of the model.

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: The paper specifies the experimental settings needed to understand the results.

   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: [No].

   Justification: The paper reports results on standard benchmarks using fixed evaluation splits and deterministic evaluation protocols, but does not include error bars, confidence intervals, or statistical significance tests.

   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: [No].

   Justification: The paper does not fully disclose the compute resources used for each experiment, due to internal infrastructure and confidentiality constraints.

   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 comply with the Code of Ethics in every respect.

   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: The paper discusses negative societal impacts.

    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: [N/A].

    Justification: The paper discusses potential downstream misuse risks, as the proposed visual reasoning capabilities could be used for unsafe applications if incorporated into deployed systems. However, this submission does not release high-risk artifacts, such as model checkpoints, deployed APIs, or other resources that would directly enable such misuse. Therefore, safeguards for responsible release of high-risk data or models are not applicable to this submission.

    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: The paper credits the original creators of the existing public assets used in the experiments and cites the corresponding papers where applicable. We do not redistribute restricted assets or use them in ways that violate their original licenses.

    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: [N/A]

    Justification: The paper does not release new 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: The paper does not involve crowdsourcing nor research with human subjects.

    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: The paper does not involve crowdsourcing or research with human subjects.

    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: The paper describes the use of LLMs/MLLMs as part of the core methodology, including their role in visual reasoning, data construction, training, and evaluation. The usage is methodological rather than limited to writing, editing, or formatting.

    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.