Exemplar2VQA: A Scalable Exemplar-Driven Visual Question Answering Generation Framework via Multi-Agent Coding

Jiayu Ying[1], **Qijian Tian**[2], **Ruijie Xu**[1], **Xinnan Zhu**[1], **Daoguo Dong**[3], **Jiachen Xu**[4], **and Xin Tan**[1†]
[1]East China Normal University [2]Shanghai Jiao Tong University
[3]Fudan University [4]Tencent
51295901118@stu.ecnu.edu.cn, xtan@cs.ecnu.edu.cn, tianqijian@sjtu.edu.cn
{ruijiex0, jcx97119, zxn.koala}@gmail.com, dgdong@fudan.edu.cn

Abstract

Advancing spatial intelligence in Multimodal Large Language Models (MLLMs) is bottlenecked by the scarcity of complex, scalable 3D question-answer (QA) data. While manual annotation is labor-intensive, directly utilizing LLMs to synthesize these QA pairs often fails due to their inherent deficiencies in spatial and geometric computation. We introduce **Exemplar2VQA**, a scalable exemplar-driven visual question answering generation framework that rapidly synthesizes large-scale spatial QA pairs in simulated environments via multi-agent coding. By equipping collaborative agents with a meticulously designed library of geometric utilities, Exemplar2VQA bypasses LLMs' spatial reasoning flaws through deterministic code execution. Crucially, the framework exhibits remarkable versatility: taking diverse static object-centric spatial query templates as exemplars, it seamlessly and autonomously scales them into massive, high-fidelity synthetic datasets. Fine-tuning Qwen2.5-VL (3B/7B) exclusively on Exemplar2VQA-generated synthetic indoor data yields significant performance improvements across various diverse benchmarks. Furthermore, its effectiveness is not limited to in-domain indoor datasets but also robustly extends to outdoor and mixed-scene benchmarks. These results establish Exemplar2VQA as a scalable and powerful paradigm for bridging the sim-to-real gap in Embodied AI. Our code is at https://github.com/yingjiayu12/Exemplar2VQA.

1 Introduction

Spatial intelligence is crucial for MLLMs in Embodied AI, yet its advancement is severely bottlenecked by a scarcity of large-scale, high-quality 3D spatial QA datasets [1]. Traditional manual annotation is prohibitively costly and unscalable. Conversely, using LLMs or VLMs to autonomously synthesize this frequently yields severe hallucinations [2]. This stems from an inherent flaw: while proficient in language, LLMs lack deterministic geometric computational abilities, making them incapable of accurately grounding physical attributes like occlusion and allocentric viewpoints [3]. While multi-agent frameworks untangle this complexity by decomposing tasks [4, 5], merely distributing the workload fails to resolve LLMs' fundamental inability to perform precise spatial arithmetic.

Consequently, synthesizing high-fidelity 3D spatial QA datasets at scale presents three formidable challenges: (1) The Data Acquisition Bottleneck: Traditional manual annotation is fundamentally unscalable and cost-prohibitive when dealing with complex, multi-view 3D geometries. (2) The

†Corresponding author.

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

Spatial Hallucination Dilemma: Directly prompting LLMs to deduce spatial attributes is highly prone to yielding logically flawed data [6, 7]. (3) The Rigidity of Existing Pipelines: Beyond the prevalent issue of hallucinations, current automated spatial generation methods are typically governed by fixed, hardcoded rules. They strictly lack the extensibility to dynamically adapt to user-defined abstract intents or flexibly generate customized QA pairs that align with diverse, specific structural formats.

To address these formidable challenges, we propose Exemplar2VQA, designed to synthesize large-scale spatial QA pairs in simulated environments (Figure 1, Figure 2) without manual annotation. At the core of this synthesis process is the utilization of spatial QA exemplars as foundational blueprints. If a user wishes to construct a specific category of spatial questions, they merely need to provide a concrete QA pair as a exemplar. This exemplar functions provides a clear schema for the pipeline by explicitly defining the desired structural format, specific object parameters, and logical comparison targets. Our pipeline then autonomously scales this user-defined intent into a massive dataset. This seamless scaling from user-defined seeds effectively resolves the rigidity of existing automated pipelines. Guided by these templates, instead of forcing LLMs to generate linguistic spatial answers directly, our framework employs a collaborative multi-agent system that translates spatial reasoning tasks into executable Python scripts. Instead of forcing a monolithic model to suffer severe cognitive overload by simultaneously juggling semantic parsing, code generation, along with review and refinement, our collaborative compartmentalizes these distinct tasks into specialized agent roles. This decoupled architecture effectively dismantles the bottleneck of single-agent systems: it prevents the context contamination and compounding errors that inevitably occur when abstract linguistic intents are mixed with mathematical execution in a single prompt. By ensuring each agent maintains a highly focused reasoning window, this framework enables targeted, module-specific self-correction, thereby drastically improving the robustness and overall execution success rate [8]. Furthermore, by leveraging interactive environmental feedback from a physics-enabled simulator [9, 10], the agents can perform physics-constrained self-correction [11–14]. This code-driven paradigm allows Exemplar2VQA to effectively bypass the inherent spatial reasoning flaws of conventional LLMs to derive reliable ground-truth answers [15–17].

To empirically validate the efficacy of our generated data, we fine-tune open-weight multimodal large language models, Qwen2.5-VL (3B and 7B) [18], exclusively on the synthetic indoor data generated by Exemplar2VQA. The experimental results demonstrate substantial improvements not only on in-domain indoor benchmarks. Without seeing any outdoor training samples, the Exemplar2VQA-enhanced models successfully transfer their acquired spatial intelligence to strictly outdoor and comprehensive mixed-scene environments. Furthermore, our approach effectively outperforms baselines fine-tuned on human-annotated real-world datasets, underscoring a highly successful sim-to-real transfer [19, 20].

In summary, the main contributions of this work are threefold:

- **An Exemplar-Driven Multi-Agent Spatial VQA Framework:** We introduce Exemplar2VQA, an end-to-end, fully automated VQA generation framework. By equipping collaborative agents with a meticulously designed library of geometric utilities and physics-constrained environmental feedback, Exemplar2VQA effectively resolves the inherent spatial computation deficiencies of LLMs through deterministic mathematical execution.
- **High Extensibility and Autonomous Scaling:** We demonstrate the exceptional versatility of Exemplar2VQA, showing that it can seamlessly adapt a broad spectrum of static object-centric spatial query exemplars from existing benchmarks and autonomously scale them into massive, high-quality synthetic datasets without manual intervention.
- **Robust Sim-to-Real and OOD Generalization:** Through extensive evaluations across diverse benchmarks, we validate that models fine-tuned solely on Exemplar2VQA's synthetic indoor data achieve extraordinary cross-domain spatial intelligence. This establishes a powerful, scalable, and cost-effective new paradigm for acquiring high-fidelity spatial data to advance Embodied AI.

2 Related Work

Spatial Understanding in Vision-Language Models has rapidly evolved from single-image reasoning [21] to complex multi-view and cross-image scenarios [3, 22–26]. To systematically diagnose

spatial intelligence, recent benchmarks such as UniQA-3D [27], ViewSpatial-Bench [28], and Multi-SpatialMLLM [29] have shifted focus towards 3D and video-based environments. Notably, Yang et al. [30] conducted comprehensive evaluations on VSI-Bench, revealing a stark reality: even advanced commercial MLLMs perform mediocrely on 3D spatial tasks. Existing spatial datasets face a critical dilemma: they either rely on fully human-curated annotations—such as MMSI-Bench proposed by Yang et al. [31], or depend on rigid, rule-based generation [32].

Automated Synthetic Data Generation has emerged as a transformative paradigm to circumvent the prohibitive costs of manual annotation. Pioneering frameworks, such as LLaVA [33] and ShareGPT4V [34], have successfully leveraged powerful models like GPT-4 to autonomously synthesize large-scale multimodal instruction-tuning data. However, while this prompting-based synthesis excels in general semantic recognition, extending it to generate complex spatial QA pairs is notoriously problematic. When directly prompted to deduce precise 3D geometric relationships, MLLMs frequently suffer from severe spatial hallucinations [35–37]. Consequently, directly prompting them to implicitly "hallucinate" physical relationships yields spatial data riddled with logical inconsistencies. Exemplar2VQA completely subverts this flawed paradigm.

Embodied Environments and Sim-to-Real Transfer have become indispensable for scalable spatial learning in Embodied AI [38–43]. 3D physics-enabled simulators, such as AI2-THOR [9] and Habitat [10], offer an unparalleled testbed for generating high-fidelity training data [44]. These platforms provide cost-effective, infinitely scalable [45, 46], and fully controllable environments where perfect scene metadata can be effortlessly extracted to construct ground-truth spatial relationships [47]. Capitalizing on this pristine simulated data, Exemplar2VQA establishes a robust and deterministic foundation for spatial reasoning.

3 Method

In this section, we present Exemplar2VQA. Designed for the autonomous synthesis of high-fidelity spatial QA datasets, Exemplar2VQA utilizes concrete spatial query exemplars as guiding seeds. We also detail the framework's overarching design, including the deterministic geometric utilities API it employs and the multi-agent collaborative pipeline that drives the dual-track synthesis process [48, 49, 11, 50].

3.1 Problem Formulation and Exemplar2VQA Overview

The fundamental objective of Exemplar2VQA is to autonomously synthesize large-scale, high-fidelity spatial QA datasets without relying on manual annotation or the hallucination-prone implicit reasoning of MLLMs. We formulate this synthesis process as a deterministic mapping. Formally, given a 3D simulated environment $\mathcal{S}$ and an abstract spatial query exemplar $\mathcal{E}$, Exemplar2VQA acts as an automated framework, denoted as Φ_{E2V} for mathematical brevity, to generate a dataset $\mathcal{D} = \{(\mathcal{V}_i, Q_i, A_i)\}_{i=1}^{N}$. Here, $\mathcal{V}_i$ denotes a set of images captured according to the user-specified camera viewpoint requirements, Q_i is the instantiated natural language question, and A_i is the ground-truth answer. Thus, the pipeline can be abstracted as $\mathcal{D} = \Phi_{\mathrm{E2V}}(\mathcal{S}, \mathcal{E})$.

To operationalize this mapping Φ_{E2V}, we introduce a logic-driven, dual-track multi-agent architecture, as illustrated in Figure 1, 2. Rather than forcing agents to implicitly guess spatial relationships, Exemplar2VQA conceptualizes them as code generators. The framework is bifurcated into two tracks.

Camera Generation Track: This track is responsible for navigating the environment $\mathcal{S}$ to capture valid and complex visual observations $\mathcal{V}_i$.

QA Generation Track: Utilizing the exact scene metadata (e.g., bounding boxes), this track programmatically derives Q_i and mathematically computes A_i.

Crucially, by abstracting spatial reasoning into parameterized programmatic execution, Exemplar2VQA bypasses the constraints of rigid, rule-based generation. As demonstrated in Figure 2, this architecture allows Exemplar2VQA to seamlessly adapt to a highly diverse spatial question taxonomy $\mathcal{E}$. From simple egocentric positioning to complex allocentric viewpoint reasoning and occlusion relationships, the framework autonomously scales these exemplars into massive, varied QA pairs, entirely eliminating the need for manual scripting.

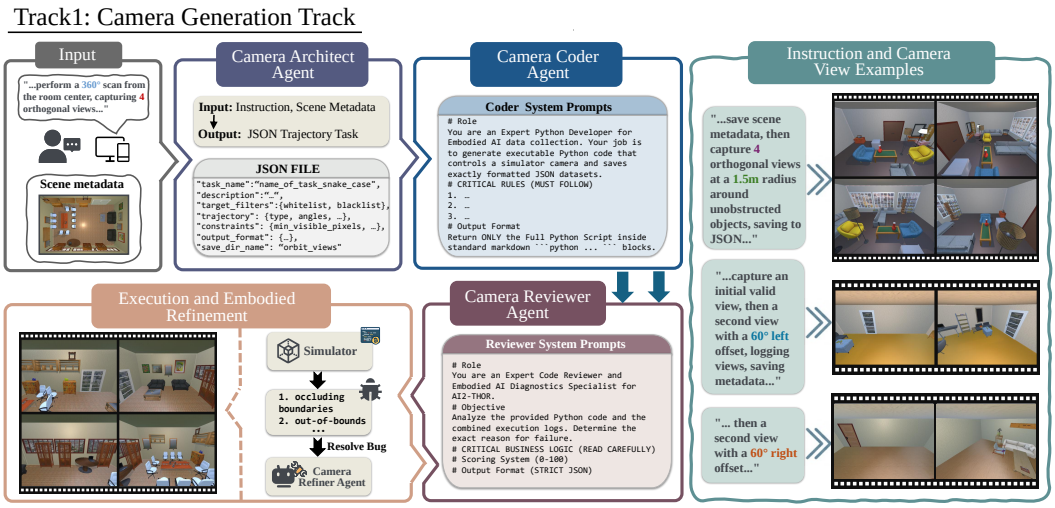


Figure 1: The camera generation track of the Exemplar2VQA. The Camera Generation Track: A four-agent pipeline (Architect, Coder, Reviewer, Refiner) interprets natural language instructions to autonomously plan trajectories and capture specific multi-view observations within the simulator.

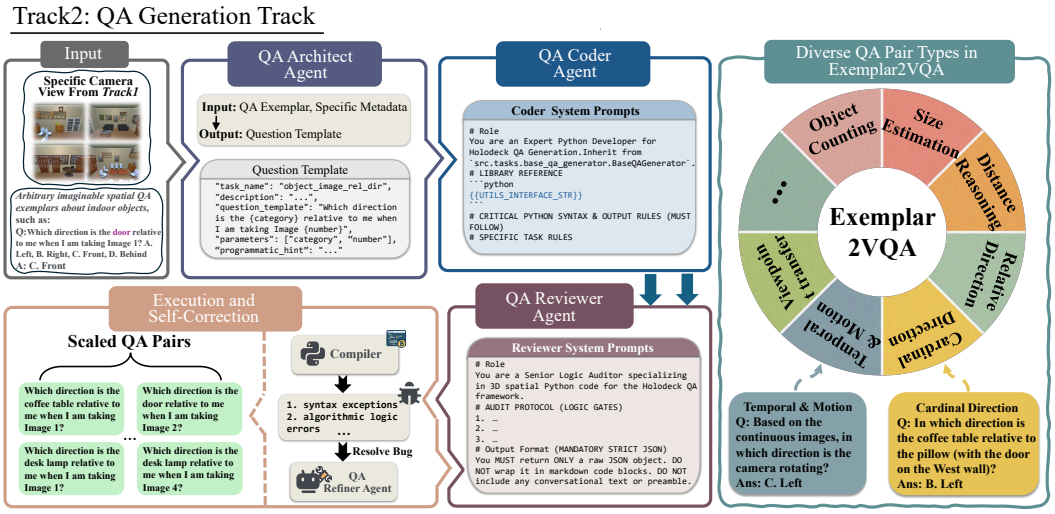


Figure 2: The QA generation track of the Exemplar2VQA. The QA Generation Track: Utilizing the captured scenes and scene metadata, an agent group adapts QA exemplars into scaled QA pairs.

3.2 Dual-Track Multi-Agent Collaborative Coding

To operationalize the deterministic mapping Φ_{E2V}, Exemplar2VQA orchestrates a logic-driven, dual-track multi-agent framework. Both tracks share a unified four-stage topology, but they operate within distinct execution environments to decouple embodied navigation from abstract spatial reasoning. All agents across both tracks are instantiated by a single foundational model: the **open-source** `Qwen3-Coder-30B-A3B-Instruct` [51]. By leveraging a highly capable yet parameter-efficient (30B) open-source model rather than relying on massive, opaque proprietary LLMs, Exemplar2VQA ensures both high accessibility and full reproducibility. The distinct agent roles are strictly enforced not by different model weights, but through specialized system prompts, compartmentalized contextual memory, and track-specific tool access. This homogeneous design maintains consistent programmatic proficiency across the pipeline while preventing the context contamination that typically plagues monolithic agents.

Track 1: Camera Generation Track. This track is responsible for autonomously navigating the simulator to capture valid and complex multi-view observations $\mathcal{V}_i$. To reliably bridge the semantic

gap between abstract human intent and deterministic physical execution, we decompose the navigation pipeline into three distinct phases handled by specialized collaborative agents.

Stage 1: Semantic Parsing. Given a natural language instruction I_{NL} (e.g., "*perform a 360° scan around the bed*"), the *CameraArchitect* agent operates as a data extraction and planning module. It translates the abstract query into a structured, standardized JSON trajectory task $\mathcal{T}_{\mathrm{json}}$. This abstraction mathematically parametrizes the required embodied actions, explicitly defining target object filters $\mathcal{F}$, trajectory typologies, viewing angles θ, and rigorous visual constraints $\mathcal{C}$ (e.g., minimum visible pixels). To handle the varying scale of indoor elements, the architect defines an adaptive radius parameter r for the camera relative to a target object o, which can be formulated as:

$$r(o) = r_{\mathrm{base}} + \frac{1}{2}\max(D_o) \tag{1}$$

where r_{base} is a predefined clearance scalar and $D_o \in \mathbb{R}^3$ represents the physical dimensions (length, width, height) of the target object's bounding box. Conversely, for tasks that do not target specific instances, this object-relative adaptive radius is bypassed.

Stage 2: Camera Code Generation. Upon receiving the parameterized configuration $\mathcal{T}_{\mathrm{json}}$, the *CameraCoder* agent acts as a geometric compiler, transforming the logical plan into an executable Python navigation script z_{cam}. This script programmatically calculates the precise sequence of camera poses $P \in SE(3)$ required to capture the scene based on the designated trajectory typology. The generated code directly invokes the underlying simulator APIs to execute the embodied navigation and capture the required multi-view visual observations $\mathcal{V}_i$.

Stage 3: Execution and Embodied Refinement. Crucially, this generated script z_{cam} interacts directly with the AI2-THOR physics engine. Due to the inherent unpredictability and geometric complexity of 3D environments, initial spatial scripts frequently trigger runtime failures, such as physical collisions with occluding boundaries or out-of-bounds rendering. When such embodied constraints are violated, the simulator returns a traceback error e_{sim}.

To systematically resolve these physical conflicts, we introduce a decoupled diagnostic framework. The *CameraReviewer* agent analyzes the execution trace and error state to diagnose the specific geometric or logical bug. Subsequently, the *CameraRefiner* agent autoregressively patches the navigation code. This iterative refinement process can be mathematically formalized as:

$$z_{\mathrm{cam}}^{(k+1)} = \mathcal{R}_{\mathrm{cam}}\left(z_{\mathrm{cam}}^{(k)}, e_{\mathrm{sim}}^{(k)}\right) \tag{2}$$

where k denotes the refinement iteration index. This physics-constrained, closed-loop simulator feedback effectively eliminates invalid observations.

Track 2: QA Generation Track. Operating downstream of Track 1, this track utilizes the precise scene metadata $\mathcal{M}$ associated with the captured observations $\mathcal{V}_i$ to mathematically derive instantiated questions Q_i and ground-truth answers A_i.

Stage 1: Semantic Parsing. Given a concrete, user-provided spatial query exemplar $\mathcal{E}$, the *QA Architect* acts as a reverse-engineering module to disentangle the underlying spatial logic. It abstracts the specific textual exemplar into a generalized, parameterized schema $\Phi_{\mathcal{E}}$. Specifically, specific object names, numerical conditions, and image identifiers within $\mathcal{E}$ are replaced by a set of abstract placeholders $\mathbf{V} = \{v_{\mathrm{cat}}, v_{\mathrm{count}}, v_{\mathrm{view}}, \ldots\}$ to formulate a highly scalable question template.

Stage 2: QA Code Generation. Guided by the schema $\Phi_{\mathcal{E}}$, the *QA Coder* synthesizes a Python script z_{qa}. It invokes our meticulously engineered library of geometric utilities (detailed in Section 3.3) to map the spatial reasoning tasks into explicit mathematical operations over the metadata $\mathcal{M}$.

Stage 3: Execution and Self-Correction. During execution within the Python runtime environment, the initial script may encounter syntax exceptions, data-type mismatches, or algorithmic logic errors, triggering a runtime error e_{py}. The *QA Reviewer* and *QA Refiner* systematically diagnose and debug these faults through an iterative autoregressive refinement loop:

$$z_{\mathrm{qa}}^{(k+1)} = \mathcal{R}_{\mathrm{qa}}\left(z_{\mathrm{qa}}^{(k)}, e_{\mathrm{py}}^{(k)}\right) \tag{3}$$

Once a structurally and logically sound script z_{qa}^{*} is compiled, it systematically iterates over the permutations of the scene metadata $\mathcal{M}$ to compute and instantiate a massive, high-fidelity dataset of QA pairs $\mathcal{D}_{\mathrm{qa}} = \{(Q_i, A_i)\}_{i=1}^{N}$.

3.3 Geometric API Orchestration

To bridge the gap between abstract spatial reasoning and physical reality, Exemplar2VQA equips the multi-agent system with a meticulously engineered library of deterministic geometric utilities. As fundamentally auto-regressive models, LLMs notoriously struggle with complex matrix transformations, and precise 3D geometric intersections, which are the primary catalysts for spatial hallucinations. Exemplar2VQA circumvents this architectural limitation by strictly offloading all rigorous geometric computations to a Python-based execution environment.

As formalized in Code 1, we expose a structured hierarchy of spatial API signatures to the code-generation agents. Rather than treating these utilities as isolated tools, the *QA Coder* synthesizes a programmatic execution plan, strictly enforcing geometric prerequisite chains. The LLM is abstracted away from continuous-space physics; its sole responsibility is topological routing and dependency orchestration. For instance, evaluating an allocentric directional relation mandates a strict order of operations: the pipeline must first invoke foundational APIs like `get_object_xz_points` to project raw 3D scene metadata into 2D coordinate footprints. Only upon completing this prerequisite projection can the higher-level relational API, `is_spatial_relation_satisfied`, perform deterministic interval intersections.

```
class Exemplar2VQA_Utils:
  """geometric API."""

  def get_object_xz_points(obj_data, inc_corners=
      False):
       # Maps 3D 'obj_data' into a 2D [X, Z]
      coordinate footprint.

  def calculate_planar_angles(vec_ref,
      vec_targets):
       # Finds the relative angle of 'vec_targets'
       from 'vec_ref'.

  def is_spatial_relation_satisfied(dir, ref, tgt
      , th=0.15):
       # Validates if 'tgt' is strictly [dir] of '
      ref'.

  def calculate_projected_distance(cam, tgt,
      angle):
       # Measures if moving at 'angle' gets 'cam'
      closer to 'tgt'.

  def is_angle_ambiguous(angles, threshold=10):
       # Rejects 'angles' near axes to prevent
      vague spatial queries.

  def cal_3d_bbox_distance_between_categories(A,
      B):
       # Finds the shortest 3D physical distance
      between groups 'A' and 'B'.
```

Code 1: **Selected** Exemplar2VQA **Geometric API**.

Furthermore, the execution plan includes built-in geometric filters to maintain the high quality of the dataset. For instance, the system automatically routes calculated results through heuristic checks like `is_angle_ambiguous`. This step acts as a mandatory safety mechanism, actively discarding confusing or borderline spatial layouts before they can be generated into final QA pairs. By conceptualizing the LLM as a high-level coordinator that orchestrates these reliable mathematical functions, Exemplar2VQA significantly mitigates the risk of spatial hallucinations.

4 Experiments

4.1 Experimental Settings

Primary Evaluation Benchmarks. We mainly evaluate our approach on MMSI-Bench [31] and VSI-Bench [30] to validate the direct performance improvements yielded by our synthesized data. Specifically, MMSI-Bench is a challenging VQA benchmark, comprising 1,000 meticulously crafted multiple-choice questions. VSI-Bench evaluates the visual-spatial intelligence of MLLMs from egocentric videos, featuring over 5,000 QA pairs derived from 288 real-world videos across diverse environments. Furthermore, comprehensive details regarding the adaptation of query exemplars from other benchmarks, along with their experimental data, are documented in Appendix A.3.

Zero-Shot Generalization Benchmarks. To assess the cross-dataset transferability of the learned spatial priors, we employ SpaCE-10 [55] and the challenging ViewSpatial-Bench [28]. Notably, while SpaCE-10 tests generalization to unseen indoor topological queries, ViewSpatial-Bench introduces highly complex, mixed indoor and outdoor scenarios. Evaluations on an additional real-world life-scene benchmark, conducted under a similar zero-shot setup, are also provided in Appendix A.3.

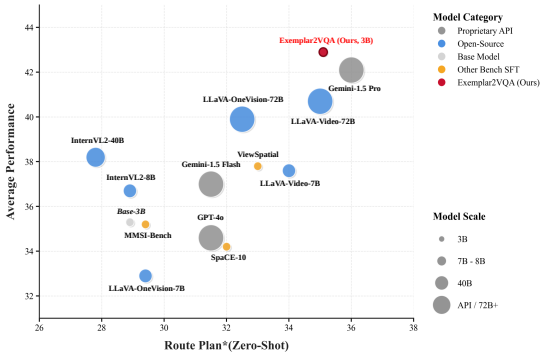
Baselines. We compare the performance of Exemplar2VQA against four primary categories of baselines: (1) proprietary MLLMs, including GPT-4o [56], Gemini-1.5 Pro [57], and Gemini-1.5 Flash [57]; (2) open-weight MLLMs, such as LLaVA-Video-72B [58], LLaVA-OneVision-72B [59], InternVL2-40B [60], InternVL3-38B [52], InternVL2.5-38B [53], Qwen2.5-VL-32B [18], LLaVA-Video-7B [58], LLaVA-OneVision-7B [59], InternVL3-8B [52], InternVL2-8B [60], and DeepSeek-VL2 [54]; (3) the corresponding base models, Qwen2.5-VL-3B [18] and Qwen2.5-VL-7B [18]; and

Table 1: **Performance comparison on the MMSI-Bench dataset**. The best results are shown in **bold**, and the second-best are underlined. Metrics marked with an asterisk (*) are evaluated in a strictly zero-shot setting. A superscript plus sign (+) indicates a performance improvement over the base Qwen2.5-VL-7B model.

Method	Overall	Cam.-Cam.	Cam.-Obj.	Cam.-Reg.	Obj.-Obj.	Obj.-Reg.	Reg.-Reg.	Meas.	Cam.	MSR.	Appr.*	Obj.*
Baseline												
InternVL3-38B [52]	26.3	21.5	23.3	25.3	20.2	35.3	33.3	**39.1**	16.2	25.8	21.2	31.6
InternVL2.5-38B [53]	27.9	18.3	22.1	34.9	22.3	**38.8**	35.8	37.5	14.9	25.3	25.8	38.2
Qwen2.5-VL-32B [18]	27.7	24.7	22.1	31.3	26.6	32.9	29.6	31.2	18.9	27.8	24.2	35.5
InternVL3-8B [52]	25.7	25.8	25.6	28.9	**31.9**	35.3	**37.0**	23.4	16.2	14.6	24.2	32.9
DeepSeek-VL2 [54]	27.1	23.7	**36.0**	22.9	**31.9**	30.6	22.2	28.1	**28.4**	28.3	15.2	26.3
Qwen2.5-VL-7B (Base) [18]	25.9	24.7	25.6	26.5	24.5	29.4	24.7	25.0	20.3	25.8	18.2	**39.5**
Qwen2.5-VL-7B (SpaCE-10) [55]	27.0	30.1	26.7	31.3	26.6	37.7	29.6	20.3	14.9	24.2	27.3	29.0
Qwen2.5-VL-7B (ViewSpatial) [28]	26.2	20.4	25.6	32.5	25.5	31.8	29.6	29.7	13.5	24.8	**30.3**	27.6
Exemplar2VQA (Ours, LoRA)	**28.2**+	**33.3**+	30.2+	33.7+	25.5+	23.5	29.6+	26.6+	23.0+	**29.3**+	27.3+	25.0
Exemplar2VQA (Ours, Full SFT)	28.0+	31.2+	31.4+	**37.4**+	25.5+	24.7	32.1+	31.3+	18.9	25.8	21.2+	29.0

Table 2: **Performance comparison on the VSI-Bench dataset**. **Left:** Quantitative results. The best results are shown in **bold**, and the second-best are underlined. Metrics marked with an asterisk (*) are evaluated in a strictly zero-shot setting. A superscript plus sign (+) indicates a performance improvement over the base Qwen2.5-VL-3B model. **Right:** Bubble chart illustrating the relationship between zero-shot Route Plan performance and Average Performance.

Method	Avg.	Rel. Dist.	Rel. Dir.	Route Plan*	Appr. Order
Proprietary Models (API)					
GPT-4o [56]	34.6	37.0	41.3	31.5	28.5
Gemini-1.5 Flash [57]	37.0	37.7	41.0	31.5	37.8
Gemini-1.5 Pro [57]	42.1	**51.3**	46.3	**36.0**	34.6
Open-source Models					
LLaVA-Video-72B [58]	40.7	42.4	36.7	35.0	**48.6**
LLaVA-OneVision-72B [59]	39.9	42.5	39.9	32.5	44.6
InternVL2-40B [60]	38.2	47.6	32.7	27.8	44.7
LLaVA-Video-7B [58]	37.6	43.5	42.4	34.0	30.6
InternVL2-8B [60]	36.7	38.0	33.4	28.9	46.4
LLaVA-OneVision-7B [59]	32.9	42.5	35.2	29.4	24.4
Qwen2.5-VL-3B (Base) [18]	35.3	34.7	42.6	28.9	35.0
Qwen2.5-VL-3B (MMSI-Bench) [31]	35.2	36.8	47.2	29.4	27.2
Qwen2.5-VL-3B (SpaCE-10) [55]	34.2	33.8	44.1	32.0	26.9
Qwen2.5-VL-3B (ViewSpatial) [28]	37.8	33.8	46.7	33.0	37.7
Exemplar2VQA (Ours)	**42.9***	43.8*	**52.1***	35.1*	40.5*



(4) spatial-specific baselines, where we fine-tune the base models on other existing human-annotated or rule-based spatial datasets (e.g., MMSI-Bench [31], SpaCE-10 [55], and ViewSpatial-Bench [28]).

Implementation Details. For evaluations on MMSI-Bench [31], we synthesized approximately 6K QA pairs mirroring the target format and fine-tuned Qwen2.5-VL-7B [18] using Low-Rank Adaptation (LoRA) [61] and Supervised Fine-Tuning (Full SFT) [62, 63]. For VSI-Bench [30], we generated roughly 10K QA pairs and applied full-parameter SFT on Qwen2.5-VL-3B [18]. To ensure balanced learning, we programmatically maintained a nearly uniform distribution across all spatial question types within these synthesized training datasets (see Appendix A.4 for detailed). Furthermore, to guarantee the reliability of the training data, a subset of the synthesized QA pairs underwent manual spot-checking prior to fine-tuning. We deliberately capped the dataset generation at these scales (6K and 10K) to conserve computational resources, as preliminary training revealed that this relatively small amount of high-quality data is already sufficient to yield substantial performance gains. In practice, the Exemplar2VQA pipeline is structurally capable of generating an infinite number of QA pairs, bounded only by the diversity of the simulated 3D rooms. Exemplar2VQA can access additional rooms via ProcTHOR [45] or Holodeck [46] (see Appendix A.2 for details). For all zero-shot evaluations on SpaCE-10 [55] and ViewSpatial-Bench [28], we consistently deployed the Qwen2.5-VL-7B model fine-tuned solely on the MMSI-Bench synthetic data. It is imperative to note that throughout all experiments, we made absolutely no modifications to the underlying model architectures. To ensure a strictly fair comparison, all baseline models fine-tuned on other existing spatial datasets were trained using the exact same hyperparameters and optimization configurations as those applied to our Exemplar2VQA-generated data. We provide comprehensive implementation details in Appendix A.1.

4.2 Performance on Primary Spatial Benchmarks

Results on MMSI-Bench. Table 1 compares Exemplar2VQA against open-source MLLMs on the multi-image benchmark MMSI-Bench. Notably, tasks requiring abstract semantic shape projection

Table 3: **Zero-shot performance comparison on the SpaCE-10 benchmark.** We evaluate the OOD generalization of our Exemplar2VQA-finetuned model against its base model and baselines fine-tuned on other spatial datasets. The best results are shown in **bold**. A superscript plus sign ($^+$) indicates a performance improvement over the base Qwen2.5-VL-7B model.

Method	Overall	Entity Quant.	Scene Quant.	Size Assess.	Obj.-Obj. Relation	Obj.-Scene Relation	Entity Presence	Function Reasoning	Spatial Planning
Qwen2.5-VL-7B (Base) [18]	33.3	32.7	**36.9**	36.9	35.3	32.3	27.6	34.2	27.5
Qwen2.5-VL-7B (MMSI-Bench) [31]	39.1	**39.4**	25.6	48.7	39.4	31.6	44.5	43.4	35.0
Qwen2.5-VL-7B (ViewSpatial) [28]	40.6	36.1	14.3	**58.4**	**47.7**	**35.6**	41.1	**51.5**	**42.5**
Exemplar2VQA (Ours)	**42.2**$^+$	38.4$^+$	24.2	55.1$^+$	45.5$^+$	35.4$^+$	**45.9**$^+$	50.3$^+$	32.5$^+$

Table 4: **Zero-shot performance comparison on the ViewSpatial-Bench dataset**. Quantitative results demonstrating the OOD generalization of our Exemplar2VQA-finetuned model against its base model and baselines fine-tuned on other spatial datasets. The best results are shown in **bold**. A superscript plus sign ($^+$) indicates a performance improvement over the base Qwen2.5-VL-7B model.

Method	Overall	Rel. Dir. (Cam)	Obj. Ori. (Cam)	Obj. Ori. (Per)	Rel. Dir. (Per)	Scene Sim. (Per)
Qwen2.5-VL-7B (Base) [18]	36.9	46.6	29.7	37.1	35.0	28.8
Qwen2.5-VL-7B (MMSI-Bench) [31]	36.5	45.3	29.7	41.4	35.6	24.7
Qwen2.5-VL-7B (SpaCE-10) [55]	36.8	44.2	24.3	48.7	38.8	23.8
Exemplar2VQA (Ours)	**45.4**$^+$	**51.1**$^+$	**30.5**$^+$	**60.6**$^+$	**41.0**$^+$	**39.5**$^+$

(*Appr.**) or dynamic temporal tracking (*Obj.**) fall outside the static, coordinate-based generative scope of our current Exemplar2VQA pipeline. Therefore, we directly evaluated these two metrics under a zero-shot setting. Key observations include: (1) Ours achieves a new state-of-the-art overall score of 28.2%, surpassing massive models like InternVL2.5-38B. Ours also outperforms the same 7B base model fine-tuned on other existing spatial datasets. (2) Ours substantially improves upon its base Qwen2.5-VL-7B model, with striking gains in complex multi-view reasoning (e.g., +8.6% in Cam.-Cam.). (3) Even in the aforementioned zero-shot scenarios, Ours surprisingly boosts performance by 9.1% on *Appr.**. (4) We observe a performance drop on *Obj.**. Fine-tuning exclusively on static spatial configurations biases the model toward static reasoning, which inevitably interferes with the base model's pre-trained temporal priors needed for dynamic tracking. (5) To verify that the gain does not stem from the LoRA adaptation itself, we additionally fine-tune the same 7B model with full SFT on the identical training set, which yields a comparable overall score.

Results on VSI-Bench. We further evaluate Exemplar2VQA on the egocentric video benchmark VSI-Bench (Table 2). Under our straightforward prompt-based SFT approach, directly training models for continuous numerical outputs yields sub-optimal results [2]. Since our primary objective is to validate the effectiveness of Exemplar2VQA-generated data for spatial reasoning, we focus exclusively on the multiple-choice subset. Furthermore, tasks such as Route Planning (*Route Plan**), fall outside the static, point-to-point generative scope of our current pipeline and are thus evaluated in a zero-shot setting. From these evaluations, we observe that: (1) Our 3B model achieves an impressive 42.9% overall, outperforming advanced proprietary models like GPT-4o and Gemini-1.5 Pro , as well as the 72B open-source LLaVA-Video. Crucially, Ours also decisively surpasses the same 3B base model fine-tuned on other existing spatial datasets. (2) Compared to its base Qwen2.5-VL-3B, Ours delivers a massive 7.6% absolute boost in overall accuracy, with exceptional enhancements in specific sub-tasks like Relative Direction. (3) Even in the zero-shot *Route Plan** scenario, Ours yields a notable 6.2% improvement over the base model. Moreover, the benefit of our synthetic data is not limited to a single backbone: fine-tuning InternVL2-2B and InternVL2-8B on the same training set yields consistent gains (see Table A3 in Appendix), suggesting transferable supervision across model scales and architectures.

4.3 Performance on Zero-Shot Generalization Benchmarks

Results on SpaCE-10. To assess zero-shot OOD generalization, we evaluate Exemplar2VQA on the SpaCE-10 benchmark (Table 3). Key observations include: (1) Ours achieves an 8.9% absolute overall improvement over the base Qwen2.5-VL-7B. Crucially, it also secures the highest overall zero-shot performance by outperforming the same base model fine-tuned on other spatial datasets. (2) Exceptional gains in physics- and geometry-heavy tasks, notably Size Assessment (+18.2%) and Function Reasoning (+16.1%). (3) The performance drop in Scene Quantification (*Scene Quant.*) is an

Table 5: **Ablation Study on Multi-Agent Configurations and Generator Capacity.** We evaluate the generation success rate (%) given 100 seed examples under various agent topologies and generator backbones. Merged roles (indicated by '+') share the same context window, whereas separated roles (separated by ',') operate as distinct agents.

Agent Configuration (Roles)	QA Generation (%)	Camera Trajectory (%)
Agent Topology		
1 Agent (All roles merged)	34.0	60.0
2 Agents (Architect, Coder+Reviewer+Refiner)	72.0	74.0
3 Agents (Architect, Coder, Reviewer+Refiner)	85.0	80.0
4 Agents (Exemplar2VQA Full: Arch., Coder, Rev., Ref.)	**92.0**	**84.0**
Generator Capacity (4 Agents)		
4 Agents (Qwen3.5-9B)	48.0	56.0
4 Agents (Qwen3-Coder-30B)	**92.0**	**84.0**

Table 6: **Comparison with Direct LLM Annotation.** We use Gemini3.5-Flash to directly annotate 10K synthetic training examples for the multiple-choice subset of VSI-Bench and fine-tune the same base model. Direct LLM annotation underperforms our code-based generation and even degrades the base model on average.

Method	Rel. Dist	Rel. Dir	Route Plan	Appr. Order	Avg.
Qwen2.5-VL-3B (Base)	34.7	42.6	28.9	35.0	35.3
Qwen2.5-VL-3B (LLM annotation)	31.7	40.4	34.5	6.3	28.2
Exemplar2VQA (Ours)	**43.8**	**52.1**	**35.1**	**40.5**	**42.9**

expected trade-off. This task requires abstract semantic grouping of "functional zones", causing slight negative transfer since Ours focuses strictly on instance-level bounding-box geometry. Nevertheless, overwhelming improvements in 7 out of 8 categories validate our approach.

Results on ViewSpatial-Bench. Extending our zero-shot OOD evaluation to ViewSpatial-Bench (Table 4), we observe: (1) Despite fine-tuning exclusively on synthetic indoor environments, ours delivers an 8.5% overall boost on this diverse benchmark featuring complex mixed and outdoor scenes. Crucially, ours also significantly outperforms the same base model fine-tuned on other spatial datasets. (2) The most striking enhancement occurs in Object View Orientation from the human perspective (*Obj. Ori. (Per)*), surging by 23.5% absolute and easily eclipsing the other spatial baselines. (3) Consistent gains in other challenging person-centric tasks, such as Scene Simulation Relative Direction (+10.7%) and Person-perspective Relative Direction (+6.0%).

4.4 Ablation Study on Multi-Agent Architectures and Generator Capacity

To validate the necessity of our decoupled multi-agent framework, we conduct an ablation study on the agent topology. We sample 100 seed examples and evaluate the generation success rate for both the QA Generation Track and the Camera Trajectory Track. Importantly, "success" strictly refers to the executed code producing mathematically and semantically correct final outputs. A comprehensive analysis of the error types is detailed in Section A.5. As presented in Table 5, we progressively merge the specialized agent roles.

Key observations include: (1) **The failure of monolithic models.** When all roles are collapsed into a single agent, the accuracy plummets to 34.0% for QA generation and 60.0% for camera navigation. This empirically confirms that forcing a single LLM to simultaneously interpret semantics, write deterministic geometric code, and resolve physics-based simulator bugs leads to severe cognitive overload. (2) **The power of isolating semantic planning.** Splitting the pipeline into 2 agents yields massive absolute surges: +38.0% in QA generation and +14.0% in camera tracking. (3) **The necessity of decoupled debugging.** Progressively separating the bug diagnosis from the code patching process steadily pushes performance to its peak (92.0% and 84.0%).

Beyond topology, we further examine the impact of generator capacity. Since Qwen3-Coder-30B is already the smallest model in the Qwen3-Coder family, we adopt Qwen3.5-9B as a smaller generator alternative under the same 4-agent pipeline. This replacement drops QA generation from 92.0% to

48.0% and camera trajectory from 84.0% to 56.0%, suggesting that generator capacity is crucial for reliably synthesizing executable scripts and valid camera trajectories.

4.5 Comparison with Direct LLM Annotation

A natural alternative to our code-based generation is to directly employ a powerful VLM to annotate QA pairs. To investigate this, we use Gemini3.5-Flash to directly label 10K synthetic training examples for the multiple-choice subset of VSI-Bench, and fine-tune the same Qwen2.5-VL-3B base model on this annotated data under identical training configurations. As shown in Table 6, direct LLM annotation is insufficient for this setting: it underperforms Exemplar2VQA by a large margin and even degrades performance relative to the base model on average, with a particularly severe drop on Appr. Order. We attribute this to the fact that multi-view spatial QA demands not only locally plausible answer prediction, but also cross-view consistency, de-duplication, and stable context tracking, which cannot be guaranteed by direct LLM/VLM labeling.

4.6 Remarks on Efficiency

Operating across two GPUs, Exemplar2VQA maintains high efficiency for large-scale data synthesis. For given abstract spatial task, the code synthesis phase requires an average overhead: the Camera Trajectory Track averages only **3 minutes** to autonomously plan, code, and resolve physics-based bugs in AI2-THOR, while the QA Generation Track averages only **2 minutes** to synthesize the deterministic mathematical scripts. Once these reusable scripts are generated, the subsequent data execution scales linearly and rapidly with the number of simulated scenes without requiring further LLM intervention. A comprehensive throughput analysis is provided in Appendix A.6.

5 Conclusion

In this work, we present Exemplar2VQA, a scalable exemplar-driven VQA generation framework that mitigates MLLMs' spatial reasoning limitations via deterministic mathematical execution. By leveraging geometric utilities and physics-constrained simulator feedback, Exemplar2VQA autonomously scales query exemplars into massive, high-fidelity spatial QA datasets. Experiments on MMSI-Bench and VSI-Bench demonstrate substantial performance gains over the base models, notably outperforming proprietary models like GPT-4o specifically on VSI-Bench. Furthermore, zero-shot evaluations on ViewSpatial-Bench and SpaCE-10 validate that spatial priors learned exclusively from synthetic indoor data effectively generalize to complex, unseen outdoor environments, offering a scalable approach to bridge the sim-to-real gap in Embodied AI.

Acknowledgements

This work was supported by the the Shanghai Committee of Science and Technology (Grant Nos. 25511103300, 25511104302, 25511102700), National Natural Science Foundation of China (Grant Nos. 62302167, U23A20343, W2521174), and the Young Elite Scientists Sponsorship Program by CAST YESS20240780.

References

[1] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[2] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas J. Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[3] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: multimodal large language models can see but not perceive. In *European Conference on Computer Vision (ECCV)*, 2024.

[4] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. Autogen: Enabling next-gen LLM applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155*, 2023.

[5] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. Metagpt: Meta programming for A multi-agent collaborative framework. In *International Conference on Learning Representations (ICLR)*, 2024.

[6] Jiayu Ying, Wenjie Sun, and Zhenye Xu. Dfw-3dvg: A dynamic feature weighting architecture guided by large language models for enhanced 3d visual grounding. In *International Conference on Virtual Reality and Visualization (ICVRV)*, 2025.

[7] Zhen Zhao, Jingqun Tang, Binghong Wu, Chunhui Lin, Shu Wei, Hao Liu, Xin Tan, Zhizhong Zhang, Can Huang, and Yuan Xie. Harmonizing visual text comprehension and generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[8] Shihao Yang, Zhicong Lu, Yong Yang, Bo Lv, Yang Shen, and Nayu Liu. Hycora: Hyper-contrastive role-adaptive learning for role-playing. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2026.

[9] Eric Kolve, Roozbeh Mottaghi, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: an interactive 3d environment for visual AI. *arXiv preprint arXiv:1712.05474*, 2017.

[10] Manolis Savva, Jitendra Malik, Devi Parikh, Dhruv Batra, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, and Vladlen Koltun. Habitat: A platform for embodied AI research. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[11] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[12] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Trans. Mach. Learn. Res.*, 2024.

[13] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[14] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Trans. Mach. Learn. Res.*, 2023.

[15] Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[16] Zhanpeng Luo, Ce Zhang, Silong Yong, Cunxi Dai, Qianwei Wang, Haoxi Ran, Guanya Shi, Katia Sycara, and Yaqi Xie. pyspatial: Generating 3d visual programs for zero-shot spatial reasoning. In *International Conference on Learning Representations (ICLR)*, 2026.

[17] Weijie Lv, Xuan Xia, and Sheng-Jun Huang. Codeact: Code adaptive compute-efficient tuning framework for code llms. *arXiv preprint arXiv:2408.02193*, 2024.

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

[19] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[20] Nguyen Hoai Thuong, Duong Duc Tin, and Le Hong Trang. Synclip-ad: Vision-language anomaly detection with contrastive learning and controlled synthetic data. In *Multi-disciplinary Trends in Artificial Intelligence (MIWAI)*, 2025.

[21] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Trans. Assoc. Comput. Linguistics*, 2023.

[22] Fei Wang, Xingyu Fu, James Y. Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, Tianyi Lorena Yan, Wenjie Jacky Mo, Hsiang-Hui Liu, Pan Lu, Chunyuan Li, Chaowei Xiao, Kai-Wei Chang, Dan Roth, Sheng Zhang, Hoifung Poon, and Muhao Chen. Muirbench: A comprehensive benchmark for robust multi-image understanding. In *International Conference on Learning Representations (ICLR)*, 2025.

[23] Qijian Tian, Jiayu Ying, Xuhong Wang, Yuan Xie, Lizhuang Ma, and Xin Tan. FLEG: feed-forward language embedded gaussian splatting from any views via compact semantic representation. In *European Conference on ComputerVision (ECCV)*, 2026.

[24] Yuzhou Ji, He Zhu, Junshu Tang, Wuyi Liu, Zhizhong Zhang, Xin Tan, and Yuan Xie. Fastlgs: Speeding up language embedded gaussians with feature grid mapping. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2025.

[25] Linfeng Li, Jian Zhao, Yuan Xie, Xin Tan, and Xuelong Li. Compassnav: Steering from path imitation to decision understanding in navigation. In *International Conference on Learning Representations (ICLR)*, 2026.

[26] Renhe Zhang, Yuyang Tan, Jingyu Gong, Zhizhong Zhang, Lizhuang Ma, Yuan Xie, and Xin Tan. S2GS: streaming semantic gaussian splatting for online scene understanding and reconstruction. *arXiv preprint arXiv:2603.14232*, 2026.

[27] Yiming Zuo, Karhan Kayan, Maggie Wang, Kevin Jeon, Jia Deng, and Thomas L. Griffiths. Towards foundation models for 3d vision: How close are we? In *International Conference on 3D Vision (3DV)*, 2025.

[28] Dingming Li, Hongxing Li, Zixuan Wang, Yuchen Yan, Hang Zhang, Siqi Chen, Guiyang Hou, Shengpei Jiang, Wenqiao Zhang, Yongliang Shen, Weiming Lu, and Yueting Zhuang. Viewspatial-bench: Evaluating multi-perspective spatial localization in vision-language models. *arXiv preprint arXiv:2505.21500*, 2025.

[29] Runsen Xu, Weiyao Wang, Hao Tang, Xingyu Chen, Xiaodong Wang, Fu-Jen Chu, Dahua Lin, Matt Feiszli, and Kevin J. Liang. Multi-spatialmllm: Multi-frame spatial understanding with multi-modal large language models. *arXiv preprint arXiv:2505.17015*, 2025.

[30] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[31] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. Mmsi-bench: A benchmark for multi-image spatial intelligence. *arXiv preprint arXiv:2505.23764*, 2025.

[32] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross B. Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[33] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[34] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on ComputerVision (ECCV)*, 2024.

[35] Chun-Peng Chang, Alain Pagani, and Didier Stricker. 3d spatial understanding in mllms: Disambiguation and evaluation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[36] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[37] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[38] Abhishek Kadian, Joanne Truong, Aaron Gokaslan, Alexander Clegg, Erik Wijmans, Stefan Lee, Manolis Savva, Sonia Chernova, and Dhruv Batra. Sim2real predictivity: Does evaluation in simulation predict real-world performance? *IEEE Robotics Autom. Lett.*, 2020.

[39] Ruijie Xu, Xinnan Zhu, Jiayu Ying, Daoguo Dong, Yuzhou Ji, and Xin Tan. MUSE: agentic 3d scene authoring via memory-grounded incremental requirement satisfaction. *arXiv preprint arXiv:2606.14168*, 2026.

[40] Xinnan Zhu, Ruijie Xu, Jiayu Ying, Daoguo Dong, Jiachen Xu, Yuan Xie, and Xin Tan. Jointedit3d: Feed-forward 3d scene editing in a unified latent space. *arXiv preprint arXiv:2606.13345*, 2026.

[41] Xin Tan, Bangwei Liu, Yicheng Bao, Qijian Tian, Zhenkun Gao, Xiongbin Wu, Zhihao Luo, Sen Wang, Yuqi Zhang, Xuhong Wang, et al. Towards safe and trustworthy embodied ai: foundations, status, and prospects. *Shanghai AI Lab., Shanghai, China, Tech. Rep.*, 2025.

[42] Sen Wang, Bangwei Liu, Zhenkun Gao, Lizhuang Ma, Xuhong Wang, Yuan Xie, and Xin Tan. Explore with long-term memory: A benchmark and multimodal llm-based reinforcement learning framework for embodied exploration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[43] Zhihao Luo, Wentao Yan, Jingyu Gong, Min Wang, Zhizhong Zhang, Xuhong Wang, Yuan Xie, and Xin Tan. Navimaster: Learning a unified policy for GUI and embodied navigation tasks. In *The Association for Computational Linguistics (ACL)*, 2026.

[44] Lechao Zhang, Haoran Xu, Jingyu Gong, Xuhong Wang, Yuan Xie, and Xin Tan. World2minecraft: Occupancy-driven simulated scenes construction. In *International Conference on Learning Representations (ICLR)*, 2026.

[45] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied AI using procedural generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[46] Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, Chris Callison-Burch, Mark Yatskar, Aniruddha Kembhavi, and Christopher Clark. Holodeck: Language guided generation of 3d embodied AI environments. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[47] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[48] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[49] Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[50] Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: program-aided language models. In *International Conference on Machine Learning (ICML)*, 2023.

[51] Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[52] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[53] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yimin Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[54] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, Xin Xie, Yuxiang You, Kai Dong, Xingkai Yu, Haowei Zhang, Liang Zhao, Yisong Wang, and Chong Ruan. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024.

[55] Ziyang Gong, Wenhao Li, Oliver Ma, Songyuan Li, Jiayi Ji, Xue Yang, Gen Luo, Junchi Yan, and Rongrong Ji. Space-10: A comprehensive benchmark for multimodal large language models in compositional spatial intelligence. *arXiv preprint arXiv:2506.07966*, 2025.

[56] OpenAI. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[57] Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, and et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[58] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data. *Trans. Mach. Learn. Res.*, 2025.

[59] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *Trans. Mach. Learn. Res.*, 2025.

[60] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, Ji Ma, Jiaqi Wang, Xiaoyi Dong, Hang Yan, Hewei Guo, Conghui He, Botian Shi, Zhenjiang Jin, Chao Xu, Bin Wang, Xingjian Wei, Wei Li, Wenjian Zhang, Bo Zhang, Pinlong Cai, Licheng Wen, Xiangchao Yan, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *Sci. China Inf. Sci.*, 2024.

[61] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

[62] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations (ICLR)*, 2022.

[63] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[64] Zihao Dongfang, Xu Zheng, Ziqiao Weng, Yuanhuiyi Lyu, Danda Pani Paudel, Luc Van Gool, Kailun Yang, and Xuming Hu. Are multimodal large language models ready for omnidirectional spatial reasoning? *arXiv preprint arXiv:2505.11907*, 2025.

[65] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[66] Fatemeh Shiri, Xiao-Yu Guo, Mona Far, Xin Yu, Reza Haf, and Yuan-Fang Li. An empirical analysis on spatial reasoning capabilities of large multimodal models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024.

A Appendix

A.1 Implementation Details

A.1.1 Training Hyperparameters

In this section, we detail the specific training configurations used to fine-tune our models on the Exemplar2VQA-generated synthetic datasets. Both models were trained utilizing the `swift` framework distributed. DeepSpeed ZeRO-2 optimization was employed in both setups to manage memory efficiency. Furthermore, to enhance instruction fine-tuning robustness, we applied NEFTune (Noise Embeddings for Fine-Tuning) with a noise alpha of 5 across all experiments. All inference evaluations were conducted on a single NVIDIA H200 GPU, while all fine-tuning procedures were distributed across four NVIDIA H200 GPUs. The dual-track Exemplar2VQA data generation pipelines were executed across two NVIDIA RTX A6000 GPUs.

Qwen2.5-VL-3B (VSI-Bench). For the downstream VSI-Bench evaluation, we performed SFT on the Qwen2.5-VL-3B base model. The training was conducted over 3 epochs with a learning rate of 1e-5, utilizing a cosine learning rate scheduler with a warmup ratio of 0.05. We applied a weight decay of 0.01. The per-device train batch size was set to 4 with 2 gradient accumulation steps, resulting in a global effective batch size of 32 across the four GPUs.

Qwen2.5-VL-7B (MMSI-Bench). For the MMSI-Bench evaluation, we utilized LoRA to fine-tune the Qwen2.5-VL-7B base model using bfloat16 (bf16) precision. The LoRA rank (r) was set to 16 with a LoRA alpha of 32. Similar to the 3B model, training spanned 3 epochs, but with a higher learning rate of 1e-4. We employed a cosine learning rate scheduler, a warmup ratio of 0.05, and a weight decay of 0.01. The per-device train batch size was configured to 2 with 4 gradient accumulation steps, also yielding a global effective batch size of 32. To accommodate the specific visual demands of the multi-image dataset, we constrained the maximum image resolution to 1,003,520 pixels.

A.1.2 Data Formatting and Training Prompts

To ensure our models learn to strictly adhere to the diverse protocols of the target benchmarks during inference, we distinctively formatted our SFT training datasets based on the input modality requirements of each task.

VSI-Bench. To accurately align our synthetic training data with the evaluation format of VSI-Bench, which fundamentally assesses visual-spatial intelligence from egocentric videos, we converted the multi-view spatial images generated by Exemplar2VQA into continuous rotation videos. Consequently, the SFT dataset was formatted into a standard conversational structure containing a single video input sequence. The training instruction explicitly guides the model to output the deterministic spatial answer along with the text content, enclosed within precise XML-style tags.

Human Prompt Template (Training):

```
<video>

[Task] Your task is to analyze the spatial arrangement of objects in the scene by
examining the provided video.

[Answer Instruction] You only need to provide *ONE* correct answer selecting from
the options listed below.  For example, if you think the correct answer is 'A.
Above' from 'A. Above B. Under C. Front D. Behind', your response should **only**
be '<answer>A. Above</answer>'.

[Question] {question_text}
```

Assistant Response:

```
<answer>{answer}</answer>
```

MMSI-Bench. Conversely, MMSI-Bench evaluates spatial intelligence across a discrete sequence of multi-view images. To accommodate this, our SFT dataset formatting dynamically incorporates multiple `<image>` tokens corresponding to the exact number of input frames. The training instruction prompt is correspondingly adjusted to emphasize the analysis of spatial arrangements and motion

across continuous discrete images, requiring the model to output solely the chosen answer letter to prevent hallucinated reasoning paths.

Human Prompt Template (Training):

```
<image> ...  (repeated N times based on sequence length)

[Task] Your task is to analyze the spatial arrangement and motion in the scene by
examining the provided continuous images.

[Answer Instruction] You only need to provide *ONE* correct answer selecting from
the options listed below.  For example, if you think the correct answer is 'A.
Above' from 'A. Above B. Under C. Front D. Behind', your response should **only**
be '<answer>A. Above</answer>'.

[Question] {question_text}
```

Assistant Response:

```
<answer>{answer}</answer>
```

This explicit, modality-aware formatting strategy effectively guides the models during the fine-tuning phase to bypass verbose implicit reasoning and predictably output answers in a deterministic, automated-evaluation-friendly structure.

A.1.3 Evaluation Prompts

During the downstream evaluation phase, we apply rigorous prompt templates to ensure the models are evaluated consistently across different benchmarks. These prompts strictly mirror the output formatting constraints established during the training phase, adapting only the modality tokens (video vs. images) to suit the respective benchmark.

Evaluation Prompt Template (VSI-Bench):

```
<video>

[Task] Your task is to analyze the spatial arrangement of objects in the scene by
examining the provided video.

[Answer Instruction] You only need to provide *ONE* correct answer selecting from
the options listed below.  For example, if you think the correct answer is 'A.
Above' from 'A. Above B. Under C. Front D. Behind', your response should **only**
be '<answer>A. Above</answer>'.

[Question] {full_question}
```

Evaluation Prompt Template (MMSI-Bench):

```
<image> ...  (repeated N times based on sequence length)

[Task] Your task is to analyze the spatial arrangement of objects in the scene by
examining the provided images.

[Answer Instruction] You only need to provide *ONE* correct answer selecting from
the options listed below.  For example, if you think the correct answer is 'A.
Above' from 'A. Above B. Under C. Front D. Behind', your response should **only**
be '<answer>A. Above</answer>'.

[Question] {full_question}
```

A.2 Infinite Generation via ProcTHOR and Holodeck

While our experiments deliberately capped the generated training datasets at 6K and 10K pairs to conserve computational resources, the Exemplar2VQA framework is fundamentally designed as a highly scalable engine. Because Exemplar2VQA relies entirely on deterministic programmatic execution interacting with simulator scene metadata, its generation capacity is bottlenecked solely

Table A1: **Performance comparison on the OSR-Bench dataset**. The best results are shown in **bold**, and the second-best are underlined. A superscript plus sign (+) indicates a performance improvement over the base Qwen2.5-VL-7B model.

Method	Avg.	Object Count	Relative Distance	Relative Direction
Baseline				
Qwen2.5-VL-72B [18]	33.5	49.8	32.5	**18.1**
LLaVA-v1.5-13B [65]	31.6	55.3	22.6	16.9
DeepSeek-VL2 [54]	25.5	57.9	5.5	13.1
Qwen2.5-VL-7B (Base) [18]	28.3	46.3	31.1	7.4
Exemplar2VQA (Ours)	**36.5**$^{+}$	**59.0**$^{+}$	**41.1**$^{+}$	9.3^{+}

by the diversity and quantity of available 3D simulated environments. To overcome the limitations of a fixed set of manually crafted scenes, Exemplar2VQA can seamlessly integrate with advanced environment generation platforms such as ProcTHOR [45] and Holodeck [46], effectively unlocking an "infinite generation" paradigm.

Scaling with ProcTHOR. ProcTHOR serves as a massive repository of procedurally generated, ready-to-use 3D environments. It provides access to tens of thousands of diverse, fully interactive houses and floorplans. Exemplar2VQA can autonomously iterate through this extensive database, continuously extracting fresh scene metadata (e.g., novel object configurations, varying room layouts, and diverse occlusions) without requiring any manual curation. This ensures that the generated spatial question-answer pairs maintain high domain diversity and rarely suffer from geometric repetition.

Customized Generation via Holodeck. For targeted domain adaptation or handling highly specific spatial scenarios, Exemplar2VQA leverages Holodeck for dynamic, language-guided environment generation. Holodeck allows the system to synthesize entirely new rooms within the AI2-THOR simulator simply by providing a natural language room definition or prompt (e.g., "A co-working space featuring a large communal desk and multiple rolling chairs arranged in a grid pattern at the room's center." or "A media room featuring a large recliner and a low TV stand arranged in a linear orientation at the room's center."). Once the prompt is processed, Holodeck autonomously populates the layout and outputs the precise scene metadata.

By utilizing ProcTHOR for broad, large-scale data expansion and Holodeck for customized, prompt-driven scene creation, Exemplar2VQA establishes a virtually unlimited, continuously expanding sandbox. This synergy guarantees an endless supply of high-fidelity spatial configurations.

A.3 Extended Experiments: Template Adaptation and QA Visualizations

In this section, we present extended experimental results to further demonstrate the exceptional versatility and OOD generalization of the Exemplar2VQA framework. As discussed in the main text, Exemplar2VQA is not confined to a predefined set of tasks; rather, it can seamlessly adapt static object-centric spatial query templates from diverse downstream benchmarks.

A.3.1 Evaluation Results on OSR-Bench

Omni-Spatial-Reasoning (OSR) Benchmarks [64]. To further investigate spatial intelligence in 360° panoramic environments, we extend our evaluation to OSR-Bench. OSR-Bench comprises over 153,000 diverse Question-Answer (QA) pairs.

Implementation Details. For the extended evaluation on OSR-Bench, we synthesized approximately 6K QA pairs utilizing the Exemplar2VQA framework, meticulously adhering to the diverse spatial query templates established by the benchmark (including object counting, relative distance, and relative direction, as visualized in Figure A1). We then fine-tuned the Qwen2.5-VL-7B model exclusively on this newly synthesized dataset. Consistent with our approach for MMSI-Bench, the fine-tuning was performed using Low-Rank Adaptation (LoRA) and distributed across four NVIDIA H200 GPUs. The specific training hyperparameters were kept strictly identical to those used for the MMSI-Bench evaluations to ensure comparative consistency. Comprehensive details regarding these training hyperparameters are provided in Appendix A.1.

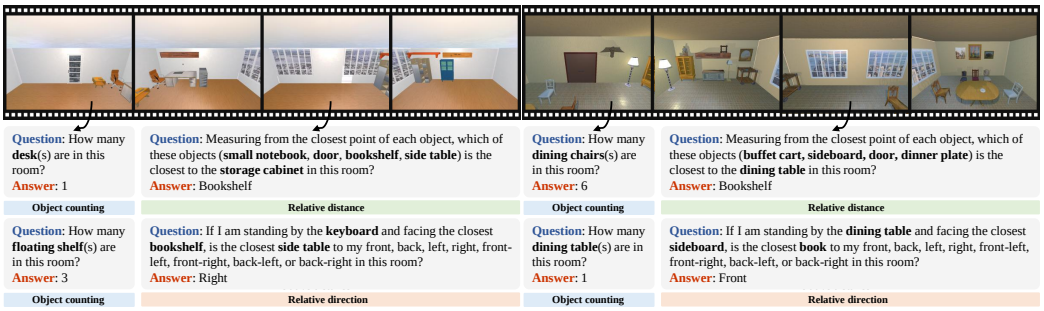


Figure A1: Visual examples of diversified spatial QA pairs and corresponding 3D scenes programmatically generated by the Exemplar2VQA framework. This visualization demonstrates the framework's versatility in adapting and scaling diverse query templates sourced from the **OSR-Bench** for **Object Counting**, **Relative Distance**, and **Relative Direction** across different generated environments to create massive and high-fidelity spatial reasoning datasets.

Table A2: **Zero-shot performance comparison on the Spatial-Obj dataset**. Quantitative results demonstrating the spatial reasoning capabilities of our Exemplar2VQA-finetuned model on one-object and two-object scenarios. The best results are shown in **bold**. A superscript plus sign ($^+$) indicates a performance improvement over the base Qwen2.5-VL-7B model.

Method	1_obj	2_obj	Overall
Qwen2.5-VL-7B (Base) [18]	83.86	62.90	69.71
Exemplar2VQA (Ours)	**84.46**$^+$	**63.85**$^+$	**70.50**$^+$

Table A3: **Generalization across backbone architectures.** We fine-tune InternVL2-2B and InternVL2-8B on the same synthetic dataset generated by Exemplar2VQA and evaluate on VSI-Bench. The best results are shown in **bold**, and the second-best are underlined. A superscript plus sign ($^+$) indicates a performance improvement over the corresponding base model.

Method	Avg.	Rel. Dist.	Rel. Dir.	Route Plan*	Appr. Order
GPT-4o [56]	34.6	37.0	41.3	31.5	28.5
Gemini-1.5 Pro [57]	42.1	**51.3**	46.3	36.0	34.6
InternVL2-2B [60]	28.2	32.1	44.1	30.4	6.3
InternVL2-8B [60]	36.7	38.0	33.4	28.9	46.4
Exemplar2VQA (Ours, InternVL2-2B)	35.6$^+$	33.6$^+$	**47.6**$^+$	36.0$^+$	25.2$^+$
Exemplar2VQA (Ours, InternVL2-8B)	**46.1**$^+$	50.4$^+$	47.2$^+$	**36.1**$^+$	**50.6**$^+$

Results on OSR-Bench. Table A1 compares Exemplar2VQA against powerful baseline MLLMs on the OSR-Bench dataset. Crucially, the training data synthesized by Exemplar2VQA consisted merely of four discrete, sparse surround-view images per scene, rather than the continuous equirectangular 360° panoramas natively evaluated in OSR-Bench. Key observations include: (1) Despite this distinct visual domain gap, Exemplar2VQA (Ours, LoRA-7B) achieves an impressive overall average of 36.5%, successfully outperforming significantly larger baseline models such as Qwen2.5-VL-72B (33.5%) and LLaVA-v1.5-13B (31.6%). Notably, it establishes top performance in both Object Count (59.0%) and Relative Distance (41.1%). (2) Exemplar2VQA delivers a massive 8.2% absolute boost in overall accuracy compared to its base Qwen2.5-VL-7B model (from 28.3% to 36.5%), with consistent enhancements across all sub-tasks, including a striking +12.7% increase in Object Count and +10.0% in Relative Distance.

While the Exemplar2VQA rendering pipeline is structurally capable of stitching and projecting continuous 360° panoramic images for training—which would naturally eliminate this modality gap and likely drive performance even higher—we deliberately omitted this step. Because our primary objective is to validate the fundamental utility and geometric accuracy of the Exemplar2VQA pipeline's automated QA generation, rather than merely engineering a state-of-the-art model tailored to a specific image format, the current substantial performance gains under this challenging zero-shot view-transfer setting are already sufficient to prove our premise.

Table A4: **Stability of MMSI-Bench results across random seeds.** We rerun the MMSI-Bench fine-tuning experiment with three different random seeds using the same training data each time, and report the mean with the standard deviation as a superscript. Metrics marked with an asterisk (*) are evaluated in a strictly zero-shot setting.

Method	Overall	Cam.-Cam.	Cam.-Obj.	Cam.-Reg.	Obj.-Obj.	Obj.-Reg.	Reg.-Reg.	Meas.	Cam.	MSR.	Appr.*	Obj.*
Exemplar2VQA	**29.33**$^{\pm 0.99}$	31.54$^{\pm 2.24}$	33.72$^{\pm 1.16}$	37.35$^{\pm 4.34}$	26.96$^{\pm 2.66}$	30.98$^{\pm 2.96}$	30.04$^{\pm 1.42}$	30.73$^{\pm 4.78}$	20.27$^{\pm 2.34}$	28.96$^{\pm 1.05}$	24.24$^{\pm 0.00}$	25.00$^{\pm 1.32}$

A.3.2 Evaluation Results on Spatial-MM Datasets

Spatial-MM Datasets [66]. To evaluate spatial reasoning capabilities in everyday environments, we utilize Spatial-MM, an **additional real-world life-scene** datasets. For our experiments, we focus exclusively on its Spatial-Obj subset. Spatial-Obj comprises 2,000 multiple-choice questions that assess spatial relationships involving one or two objects within an image. The dataset categorizes these spatial configurations into distinct visual patterns, including object localization, orientation and direction, viewpoints, and positional and relational context. Furthermore, the questions are structured to evaluate spatial relationships from both the standard camera perspective and the human perspective within the image.

Implementation Details. For the evaluation on the Spatial-MM benchmark, we adopted a strictly zero-shot setting, consistent with our methodology for the SpaCE-10 and ViewSpatial-Bench evaluations. Specifically, we deployed the Qwen2.5-VL-7B model that was fine-tuned exclusively on the synthetic data generated for MMSI-Bench. No additional, dataset-specific training or parameter updates were performed using the Spatial-MM data. This rigorous zero-shot setup allows us to genuinely assess the cross-dataset transferability of the spatial priors learned from the Exemplar2VQA pipeline. Comprehensive implementation details regarding the acquisition of this deployed model, including the specific training hyperparameters, are detailed in Appendix A.1.

Results on Spatial-Obj Subset. Table A2 details the zero-shot performance of Exemplar2VQA on the Spatial-Obj datasets. Crucially, while the Exemplar2VQA model was fine-tuned exclusively on synthetic indoor environments, Spatial-Obj evaluates spatial reasoning across diverse, real-world life scenes. Key observations include: (1) Despite this challenging sim-to-real domain gap, Exemplar2VQA (Ours, LoRA-7B) consistently outperforms the Qwen2.5-VL-7B base model across all evaluated configurations. (2) Exemplar2VQA achieves steady improvements in both single-object (*1_obj*, +0.60% absolute) and two-object (*2_obj*, +0.95% absolute) spatial scenarios, yielding an overall accuracy boost to 70.50%.

While the absolute numerical gains may appear modest compared to our in-domain evaluations, they are highly meaningful within this strict zero-shot context. The base model already exhibits a remarkably high performance baseline on this specific dataset (approaching 84% on *1_obj*, indicating near-saturation). The ability of Exemplar2VQA to squeeze out further improvements on unseen, complex real-world photographs—relying strictly on geometric priors learned from synthetic multi-view data—confirms that our automated generation framework injects robust, universally transferable spatial intelligence rather than simply overfitting to the synthetic domain.

A.3.3 Stability Across Random Seeds

To examine the stability of our MMSI-Bench results, we rerun the fine-tuning experiment with three different random seeds, using the same training data each time. As shown in Table A4, the repeated runs suggest that the reported gains are stable across seeds, although the magnitude of improvement varies across sub-categories.

A.4 Arbitrary Control and Analysis of QA Category Distributions

The Bottleneck of Existing Benchmarks. While recent benchmarks have significantly advanced the evaluation of spatial intelligence in Multimodal Large Language Models (MLLMs), relying on human-annotated datasets inevitably introduces static and imbalanced category distributions. Due to the inherent preferences of human annotators and the varying difficulty of question formulation, conventional datasets often exhibit a heavy bias towards straightforward spatial queries (e.g., basic relative direction or object recognition). When MLLMs are fine-tuned on such skewed distributions,

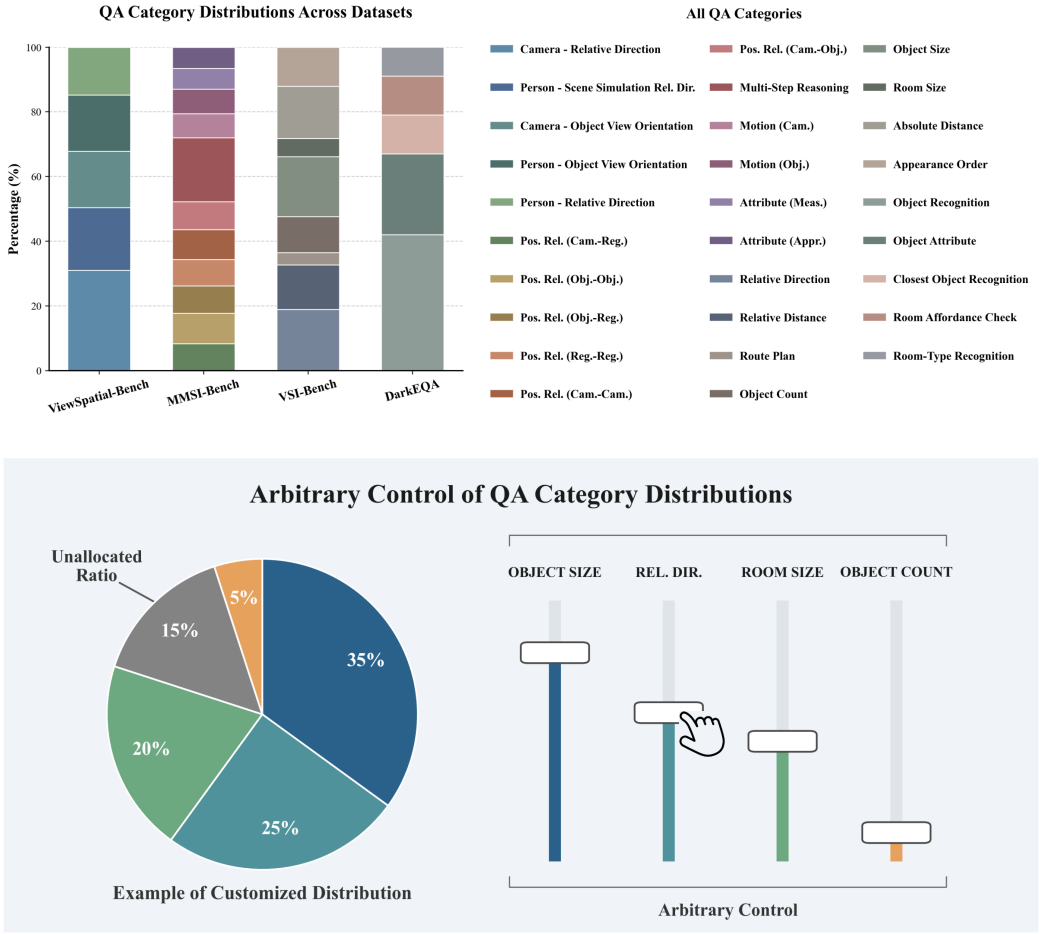


Figure A2: **Customizable QA category generation versus static benchmark distributions. (Top)** Existing spatial reasoning datasets (e.g., ViewSpatial-Bench, MMSI-Bench) exhibit fixed and inherently imbalanced QA category distributions. **(Bottom)** In contrast, our proposed pipeline enables arbitrary, programmatic control over the generation ratios. Driven by a theoretically infinite generation capacity—bounded only by the diversity of available 3D scenes—our framework allows users to dynamically configure the distribution of specific spatial queries. This paradigm effectively overcomes the static distribution biases prevalent in traditional human-annotated benchmarks.

Table A5: **Performance comparison on the four numerical tasks of VSI-Bench**. The best results are shown in **bold**, and the second-best are underlined. A superscript plus sign ($^+$) indicates a performance improvement over the base Qwen2.5-VL-3B model.

Method	Obj. Count	Abs. Dist	Obj. Size	Room Size	Avg.
GPT-4o [56]	**46.2**	5.3	43.8	**38.2**	33.4
Qwen2.5-VL-3B (Base) [18]	21.9	18.4	22.4	27.2	22.5
Exemplar2VQA (Ours)	33.6$^+$	**29.7**$^+$	**58.8**$^+$	27.4$^+$	**37.4**$^+$

they tend to develop strong preference biases, resulting in degraded generalization performance when confronted with rare or out-of-distribution spatial configurations.

Exemplar2VQA's Arbitrary Control Paradigm. To overcome this critical bottleneck, Exemplar2VQA fundamentally shifts the paradigm from static dataset curation to dynamic, programmable data synthesis. Because Exemplar2VQA relies on a deterministic multi-agent coding framework interacting with scene metadata, it functions as an autonomous generation engine rather than a

fixed dataset. Driven by a theoretically infinite generation capacity—bounded only by the diversity and quantity of available 3D simulated scenes (as detailed in Appendix A.2 via integrations with ProcTHOR and Holodeck)—Exemplar2VQA allows researchers to programmatically define the exact synthesis ratios across any spatial query categories.

Distribution Comparison and Visualization. As illustrated in Figure A2 (Top), existing leading benchmarks (such as ViewSpatial-Bench, MMSI-Bench, and VSI-Bench) are constrained by rigid QA category proportions, with certain basic spatial categories dominating the visual evaluations. In stark contrast, the bottom panel of Figure A2 conceptually visualizes Exemplar2VQA's arbitrary control mechanism through an intuitive "slider" metaphor. Mechanistically, rather than relying on complex data augmentation or rebalancing algorithms, Exemplar2VQA achieves this flexibility through sheer scale. Because the pipeline synthesizes an overwhelmingly large and diverse pool of candidate QA pairs across thousands of simulated rooms, deriving a dataset with any exact, customized category distribution is as straightforward as performing targeted random sampling from this massive pool. This effectively amplifies the presence of scarce spatial queries, effortlessly circumventing the data imbalance bottleneck.

Implications for MLLM Spatial Intelligence. This unprecedented flexibility offers profound implications for advancing the 3D spatial reasoning capabilities of Multimodal Large Language Models (MLLMs):

1. Targeted Remediation for Spatial Hallucinations: When diagnostic benchmarks reveal that an MLLM struggles with specific geometric concepts (e.g., exhibiting severe spatial hallucinations regarding 3D occlusion, allocentric viewpoints, or relative depth ordering), researchers are no longer bound by the prohibitive costs of manual data collection. Exemplar2VQA can instantly synthesize targeted diagnostic datasets, focused supervised fine-tuning.

2. Curriculum Learning for 3D Cognition: Exemplar2VQA's programmable distribution naturally facilitates curriculum learning for spatial intelligence. Model training pipelines can be dynamically structured to mimic progressive cognitive development—initiating with a distribution rich in basic instance recognition and simple egocentric relative directions, before smoothly transitioning into highly complex, allocentric 3D reasoning and multi-view geometric transformations. This progressive, data-driven approach yields a much more robust and compositionally sound spatial representation within the MLLM.

A.5 Comprehensive Error Analysis

To gain deeper insights into the operational bottlenecks of the Exemplar2VQA pipeline, we conduct a comprehensive failure mode analysis. We categorize the types of errors encountered during the generation process and compare their distribution between the monolithic (1 Agent) and multi-agent (4 Agents) configurations. It is crucial to note that, as established in Section 4.4, the absolute number of failures drops significantly under the multi-agent framework; therefore, the following analysis examines the proportion of remaining failure modes.

QA Generation Track Errors. The QA Generation Track relies on mathematical computations over scene metadata. We classify its failures into three distinct categories: *Absence of Expected QA Pairs*, where the agent fails to output any valid QA pairs—despite manual inspection confirming that the scene is indeed capable of generating the expected QA pairs; *Code Execution Failure*, occurring when the generated Python script crashes due to syntax errors, infinite loops, or incorrect API invocations; and *Incorrect QA Pair Generation*, where the code executes successfully but the resulting QA pairs contain logical flaws or mathematically incorrect ground-truth answers. In the monolithic setup, the dominant error is the Absence of Expected QA Pairs (47.0%), followed closely by Code Execution Failure (30.3%). This indicates that forcing a single LLM to simultaneously act as a semantic planner, a geometric coder, and a syntax debugger leads to severe cognitive overload. Transitioning to the multi-agent setup, the distribution shifts significantly. The proportion of Code Execution Failures decreases to 25.0%, and Absence of Expected QA Pairs drops to 37.5%.

Camera Trajectory Generation Track Errors. The Camera Trajectory Generation Track interacts directly with the AI2-THOR physics engine. We categorize its failures into three types: *Trajectory Semantic Misalignment*, where the generated trajectory executes successfully but fails to capture the specific visual observations requested by the abstract instruction; *Code Execution Failure*, which involves Python syntax errors or logical breakdowns in the navigation loop; and *FoV Occlusion &*

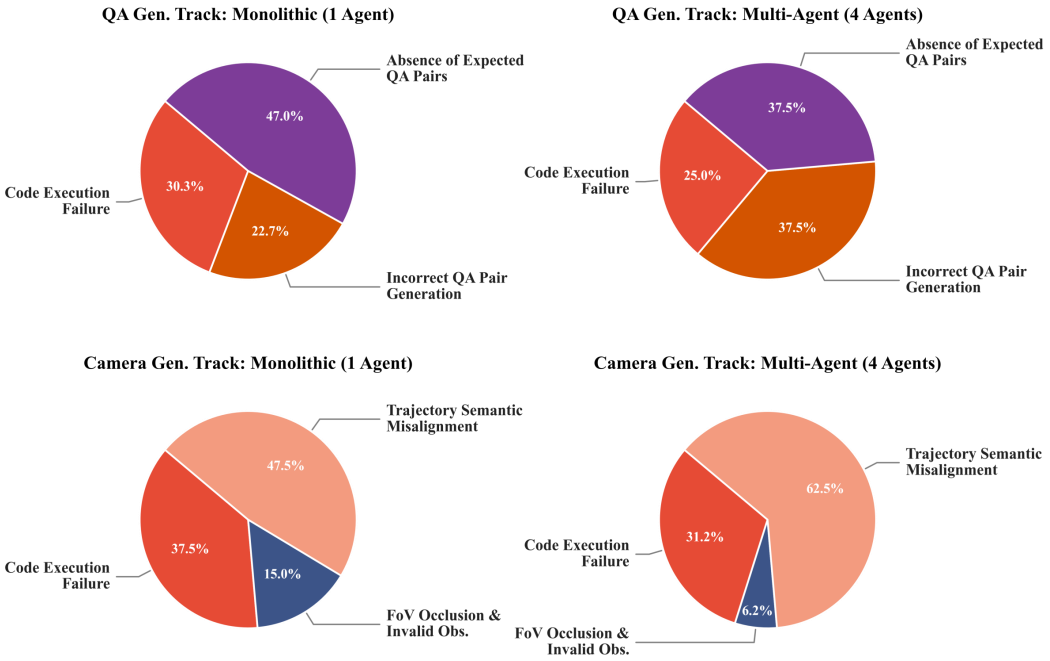


Figure A3: Distribution of error types in the Exemplar2VQA pipeline.

Invalid Observations, where the agent successfully moves, but the captured frames are invalid due to physical constraints such as staring directly into a wall, heavy occlusion, or out-of-bounds rendering. In the monolithic setup, embodied physical constraints are a major bottleneck, with Code Execution Failures (37.5%) and FoV Occlusion & Invalid Observations (15.0%) collectively accounting for over half of the errors. Conversely, the multi-agent setup demonstrates a profound ability to resolve these embodied physical errors. The closed-loop physics feedback provided to the Refiner agent drastically shrinks the proportion of FoV Occlusion down to a mere 6.2%, and Code Execution Failures drop to 31.2%. As the pipeline reliably solves physical collisions and syntax crashes, the remaining errors become overwhelmingly dominated by Trajectory Semantic Misalignment (62.5%). This highlights a critical frontier for Embodied AI: while multi-agent coding with physics feedback can guarantee a safe and executable trajectory, aligning an agent's continuous spatial navigation perfectly with nuanced, abstract human semantic intent remains an inherently challenging problem.

A.6 Detailed Efficiency and Throughput Analysis

In Section 4.6, we briefly discussed the efficiency of the Exemplar2VQA framework. A critical advantage of Exemplar2VQA's multi-agent coding paradigm is the structural decoupling of *LLM-driven Code Synthesis* (Phase 1) from *Simulator-driven Data Execution* (Phase 2). This separation ensures that the computationally expensive LLM inference acts strictly as a one-time initial overhead per spatial query template, while the subsequent large-scale dataset generation operates as a highly scalable programmatic execution.

Phase 1: Code Synthesis. This phase relies on the dual-track multi-agent system to interpret semantic templates and output bug-free Python scripts.

- **Camera Trajectory Track:** It takes an average of **3 minutes** per novel spatial task to complete the entire pipeline: semantic planning, code generation, and iterative interactive debugging within the AI2-THOR physics engine (e.g., resolving wall collisions or out-of-bounds errors).
- **QA Generation Track:** Because this track relies on mathematical computations over scene metadata rather than embodied navigation, the agents can typically synthesize and debug the optimal geometric QA code in approximately **2 minute** per task.

Phase 2: Data Execution. Once the verified Python scripts (z'_{cam} and z_{qa}) are synthesized, Exemplar2VQA completely bypasses the LLMs. The data generation process scales seamlessly across varying numbers of 3D scenes.

- **Camera Rendering Execution:** Executing the generated camera trajectory script to navigate, render, and save the required multi-view images takes approximately **15 seconds** per 3D scene.
- **QA Pair Instantiation:** To maximize the yield of generated QA pairs, the LLM-synthesized scripts frequently employ brute-force enumeration algorithms. While this can theoretically result in high polynomial complexities (e.g., up to $\mathcal{O}(M^4)$ for complex multi-object spatial relationships, where M is the number of objects), the actual number of observable objects per room is typically small ($M \approx 20$). Executing the code to instantiate QA pairs across **100 distinct scenes requires only about 25 seconds** in total.

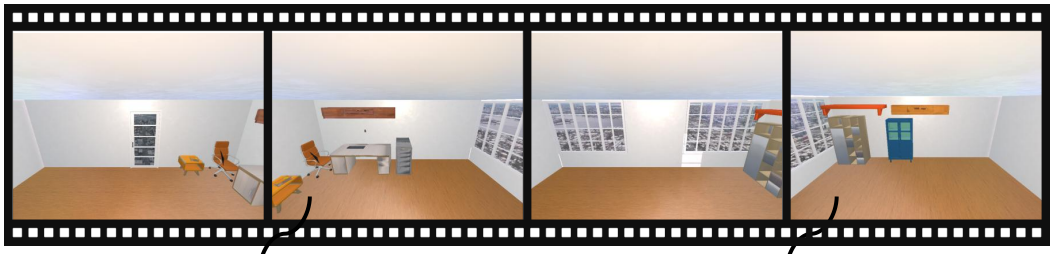
In summary, this bipartite architecture allows Exemplar2VQA to effectively leverage the intelligence of LLMs to establish a robust programmatic foundation, subsequently relying on pure CPU/GPU arithmetic to rapidly scale the spatial dataset to theoretically infinite bounds.

A.7 Limitations

While Exemplar2VQA establishes a robust paradigm for scalable spatial QA synthesis, its current scope is predominantly optimized for static, object-centric configurations, leaving complex dynamic spatiotemporal events (e.g., continuous motion tracking) for future exploration. Furthermore, the generation framework currently relies on explicit 3D scene metadata extracted from simulators; directly ingesting raw real-world photographs to autonomously synthesize QA pairs will require integrating advanced upstream models. Finally, because the pipeline relies on multi-agent code execution to bypass spatial hallucinations, its absolute success rate remains inherently bounded by the coding capabilities of the foundational LLM, particularly when navigating highly convoluted algorithmic edge cases or physics engine glitches.

A.8 Qualitative Examples of Synthesized Spatial QA Pairs

To further illustrate the versatility and generative fidelity of the Exemplar2VQA framework, this section provides a comprehensive gallery of qualitative examples. The following figures showcase a diverse array of 3D simulated environments alongside their corresponding, programmatically synthesized spatial QA pairs. These visualizations highlight Exemplar2VQA's capability to accurately instantiate a wide spectrum of complex geometric queries.



Question: What is the size of this **room** (in square meters)? \nIf multiple rooms are shown, estimate the size of the combined space. A. 62.7, B. 54.1, C. 42.6, D. 11.1
Answer: C

Question: If I am standing by the **keyboard** and facing the closest **bookshelf**, is the closest **side table** to my front, back, left, right, front-left, front-right, back-left, or back-right in this room?
Answer: Right

Question: When entering the room through the door, where is the **desk** located relative to your position? A. To the front left, B. To the right, C. To the left, D. To the front right
Answer: A

Question: Which direction is the **desk** relative to me when I am taking **Image 1**? A. Left, B. Right, C. Front, D. Behind
Answer: B

Question: What is the length of the longest dimension (length, width, or height) of the **side table**, measured in centimeters?
Answer: 63

Question: What is the length of the longest dimension (length, width, or height) of the **door**, measured in centimeters? A. 328, B. 209, C. 132, D. 87
Answer: B

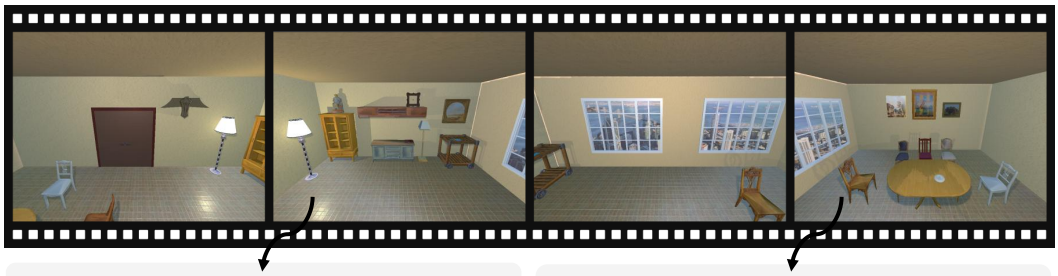
Question: How many **desk**(s) are in this room?
Answer: 1

Question: How many **door**(s) are in this room?
Answer: 1

Question: When entering the room through the door, where is **the storage cabinet** located relative to your position? A. To the front left, B. To the front right, C. To the right, D. Directly in front
Answer: B

Question: Are there at least 1 **keyboard**(s) in this room?
Answer: Yes

Figure A4: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.



Question: When you took Image 1, where was the **camera** for **Image 2**, relative to you? A. Left, B. Back, C. Front, D. Right
Answer: Right

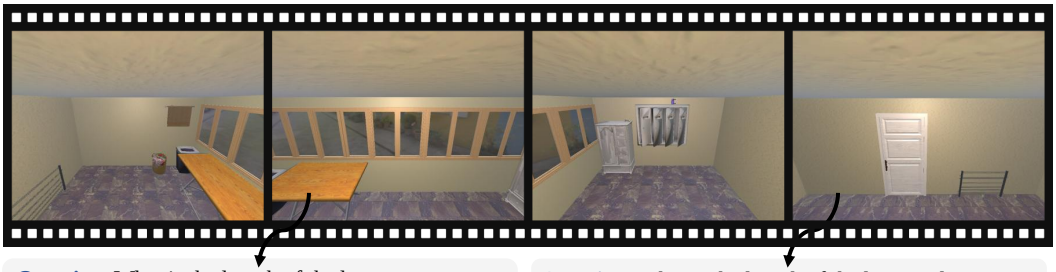
Question: Are there fewer **floor lamp** than **wall art**? Please reply with a \"yes\" or \"no\" only.
Answer: yes

Question: Are there fewer **dining table** than **floor lamp**? Please reply with a \"yes\" or \"no\" only.
Answer: yes

Question: Is the distance between the **dining table** and the **book** greater than the distance between the **sideboard** and the **wall mirror**? Please reply with a \"yes\" or \"no\" only.
Answer: yes

Question: If I am standing by the **buffet cart** and facing the **sideboard**, is the **dinner plate** to the left or the right of the **sideboard**?\nAssume \"standing by\" means I am positioned at the center of the buffet cart. A. right, B. left
Answer: B

Figure A5: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.



Question: What is the length of the longest dimension (length, width, or height) of the **laundry basket dirty**, measured in centimeters? A. 92, B. 58, C. 39, D. 74
Answer: B

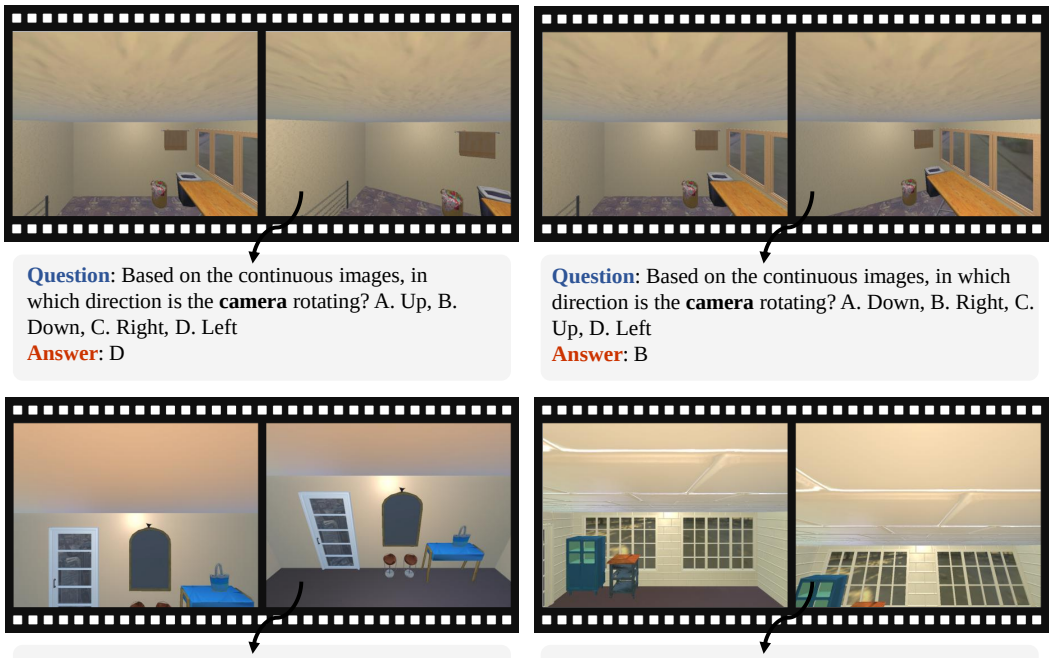
Question: What is the length of the longest dimension (length, width, or height) of the **folded blanket**, measured in centimeters?
Answer: 26

Question: If I am standing by the **laundry basket dirty** and facing the **laundry basket clean**, is the **drying rack** to my front-left, front-right, back-left, or back-right?\nAssume \"standing by\" means I am positioned at the center of the laundry basket dirty.\nThe directions refer to the quadrants of a Cartesian plane (if I am standing at the origin and facing along the positive y-axis). A. front-right, B. back-left, C. front-left, D. back-right"
Answer: D

Question: How many **laundry basket dirty**(s) are in this room?
Answer: 1

Question: If I am standing by the **door** and facing the **folding table**, is the **floor storage cabinet** to my left, right, or back?\nAssume \"standing by\" means I am positioned at the center of the door.\nAn object is to my back if I would have to turn at least 135 degrees in order to face it. A. back, B. left, C. right
Answer: C

Figure A6: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.



Question: Based on the continuous images, in which direction is the **camera** rotating? A. Up, B. Down, C. Right, D. Left
Answer: D

Question: Based on the continuous images, in which direction is the **camera** rotating? A. Down, B. Right, C. Up, D. Left
Answer: B

Question: Based on the continuous images, in which direction is the **camera** rotating? A. Up, B. Down, C. Right, D. Left
Answer: B

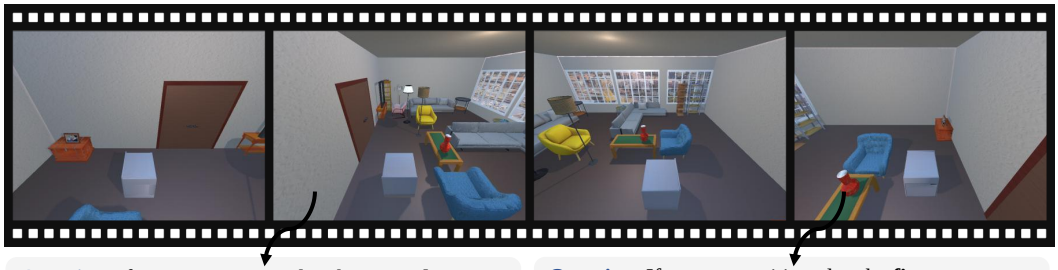
Question: Based on the continuous images, in which direction is the **camera** rotating? A. Down, B. Right, C. Up, D. Left
Answer: C

Figure A7: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.



Question: If you are positioned at the **first viewpoint**, what is located entirely to the behind of the **coffee table** from where you stand? A. console table, B. ottoman, C. sofa, D. picture frame
Answer: C

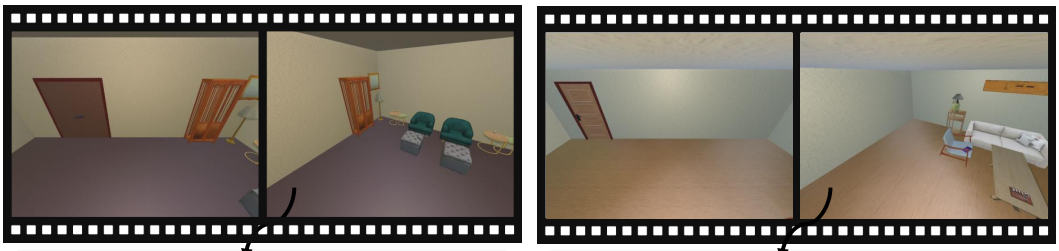
Question: If you are positioned at the **fourth viewpoint**, what is located entirely to the left of the **coffee table** from where you stand? "A. ottoman, B. sofa, C. picture frame, D. console table
Answer: C



Question: If you are positioned at the **second viewpoint**, what is located entirely to the left of the **ottoman** from where you stand? A. vase, B. picture frame, C. armchair, D. side table
Answer: B

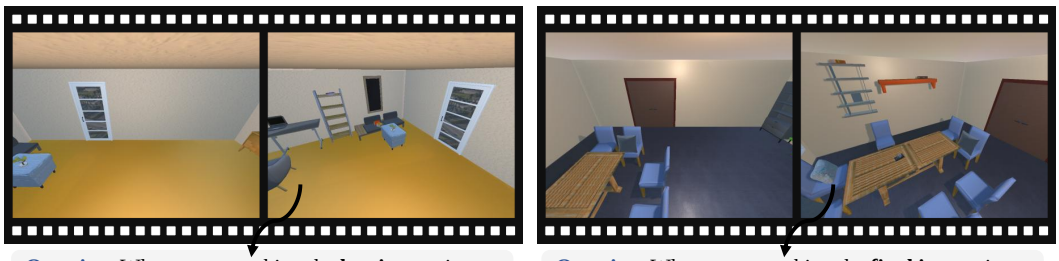
Question: If you are positioned at the **first viewpoint**, what is located entirely to the front of the **ottoman** from where you stand? A. coffee table, B. armchair, C. sofa, D. picture frame
Answer: D

Figure A8: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.



Question: When you are taking the **initial image**, in which direction is the **wall mirror** located relative to you? A. Right, B. Rear, C. Left, D. Front
Answer: A

Question: When you are taking the **second image**, in which direction is the **coffee table** located relative to you? A. Rear, B. Left, C. Right, D. Front
Answer: C



Question: When you are taking the **last image**, in which direction is the **laptop** located relative to you? A. Right, B. Rear, C. Front, D. Left
Answer: D

Question: When you are taking the **final image**, in which direction is the **dining table** located relative to you? A. Right, B. Left, C. Front, D. Rear
Answer: C

Figure A9: Additional qualitative examples of diverse spatial reasoning QA pairs autonomously synthesized across simulated environments.

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: The claims made in the abstract and introduction accurately reflect the paper's scope and contributions.

   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 this work in Appendix A.7.

   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: This paper focuses on the empirical design, implementation, and evaluation of the Exemplar2VQA data generation framework. It does not introduce any theoretical results, mathematical theorems, or formal proofs; therefore, this question is not applicable.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We fully disclose all information necessary to reproduce the main experimental results, including but not limited to the details provided in Sections 3, 4, and Appendix A.1.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: The code of this work will be open-sourced upon publication of the paper.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: We specify all the training and test details in Section 4 and Appendix A.1.

   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 primary contribution of this work is the proposal of the Exemplar2VQA data generation pipeline. Our extensive evaluations across multiple diverse benchmarks demonstrate that models fine-tuned on Exemplar2VQA-generated data achieve substantial and consistent performance improvements over baselines (e.g., an absolute gain of +7.6% on VSI-Bench and +8.9% on SpaCE-10). The significant magnitude of these gains across various tasks provides strong empirical evidence of the framework's efficacy, well beyond marginal statistical fluctuations. Given this robust multi-benchmark validation, coupled with the computational resources required to repeatedly fine-tune Multimodal Large Language Models (3B and 7B), performing multiple independent trials with different random seeds to compute error bars was not conducted.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Computational resources are described in detail in Section 4 and Appendix A.1.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have thoroughly read the NeurIPS Code of Ethics and ensured compliance in all aspects of our research.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

Justification: We present a discussion of the broader impact and societal implications of this work in Appendix A.4.

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: Not involved in misusing.

    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: All existing assets utilized in this research, including 3D simulators, pre-trained models, and evaluation benchmarks, are properly cited in the references.

    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: This paper does not introduce new assets (except for data and code, which will be open-sourced upon publication of the paper).

    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: This paper does not involve crowdsourcing experiments or 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: This paper does not involve crowdsourcing experiments 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 core methodology of this research relies on a multi-agent framework powered by Large Language Models. We explicitly detail the usage of the LLM in Section 3, specifying the deployment of the open-source Qwen3-Coder-30B-A3B-Instruct model. We thoroughly describe how it is utilized across distinct agent roles (Architect, Coder, Reviewer, Refiner) to programmatically generate code and autonomously synthesize the spatial QA datasets.

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.