SPARK: Path-Aware Experiential Self-Evolution for VLMs Spatiotemporal Reasoning

Chengzhengxu Li[1], **Xiaoming Liu**[1,*], **Zhaohan Zhang**[2], **Yu Lan**[1], **Zicheng Zhao**[1], **Bingxiang Wang**[1], **Cong Wang**[3], **Chao Shen**[1]
[1]Faculty of Electronic and Information Engineering, Xi'an Jiaotong University
[2]Queen Mary University of London, UK [3]City University of Hong Kong
[*] Corresponding author
{czx.li, zzc0109}@stu.xjtu.edu.cn
{xm.liu, ylan2020, chaoshen}@xjtu.edu.cn
zhaohan.zhang@qmul.ac.uk congwang@cityu.edu.hk

Abstract

While vision-language models (VLMs) perform well on general tasks, their spatiotemporal reasoning remains limited. Existing distillation and Reinforcement Learning (RL) methods are often computationally expensive and poorly generalizable. Recent studies suggest that model self-improvement by experience distilled from success and failure trajectories is a promising direction. However, in VLM tasks where reasoning must be grounded in visual information, valuable signals of reasoning quality arise not only from final outcomes, but also from how faithfully the reasoning path interacts with visual information. We examine the dependency between reasoning paths and visual information, and observe that high-quality paths exhibit stronger visual anchoring and greater visual dependency. Building on this observation, we introduce the **S**patiotemporal **P**ath-**A**ware **R**easoning paradigm based on experiential **K**nowledge (SPARK) that encourages the model to explore and exploit valuable experiences derived from paths with subtle quality differences. To enable diverse and efficient path exploration, we integrate Monte Carlo Tree Search (MCTS) with fine-grained visual rewards and submodular optimization, allowing the search process to prune redundant branches while preserving critical reasoning paths. We further present a training-free experience construction mechanism that converts contrastive pairs of distinct-value paths into structured experiences, which are dynamically injected into the VLM through similarity-based retrieval during inference. Extensive experiments demonstrate that SPARK enables Qwen3.5-9B to achieve an average accuracy of 61.17% across three benchmarks, outperforming Gemini-3-Pro (57.20%). The code is available at: https://github.com/czx-li/spark

1 Introduction

Humans naturally possess spatiotemporal intelligence, a cognitive capacity that transcends the perception of static 3D geometric forms by integrating time as an inherent dimension of comprehension [1, 2, 3]. This ability to reason about the world within a spatiotemporal framework, which unites three-dimensional space with the progression of time, allows us to navigate and act effectively in dynamically changing environments using nothing but visual inputs [4]. Such spatiotemporal reasoning ability is crucial for interactive artificial intelligence systems, including robotics [5], virtual reality [6], and embodied agents [7], as navigating dynamic scenarios requires continuous comprehension of evolving spatial relationships over time. While Visual Language Models (VLMs) have demonstrated remarkable general intelligence across various tasks, such as mathematical reasoning [8], code

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

generation [9], and image-text grounding [10], their capabilities in spatiotemporal understanding and reasoning remain largely underdeveloped [11, 12]. To enhance the spatiotemporal capabilities of VLMs, existing paradigms predominantly rely on large-scale expert distillation [13, 14, 15, 16] or Reinforcement Learning (RL) [17, 18, 19]. While effective, these training-based methods are often hindered by high data costs, sparse rewards, and inherent biases in teacher models [20, 21, 22]. More importantly, the knowledge acquired is implicitly parameterized and non-transferable, resulting in poor scalability and a lack of transparency in reasoning. More definitions to this task refer to Appendix A.1.

Instead of relying on repetitive parameter training, recent works [23, 24, 25, 26] have explored experience-driven self-evolution paradigm, where models extract reusable lessons from failed paths and use them as references for future actions. This experiential paradigm provides a more lightweight, interpretable, and adaptable mechanism for improving model decision-making [27, 8]. We analogize the model's experience acquisition process to a student learning from a constructed mistake notebook [28, 29]. In this process, the value of the acquired experience is fundamentally determined by the quality of the notebook: only when failure details are finely perceived can they provide useful guidance for future decision-making. However, existing works on experience extraction is primarily outcome-based, where path quality is judged by the correctness of the final decision [30, 8, 31]. Such evaluation falls short in capturing fine-grained differences in reasoning quality, especially when paths share the same final outcome but differ substantially in their reasoning and decision process [32]. To distinguish reasoning paths of varying quality beyond accuracy, we probe the question: *What properties do high-quality reasoning paths exhibit for visual input?* Through pilot experiments in §3, we obtain the following finding ($\mathcal{F}$): **The reasoning paths with better performance exhibit stronger visual anchoring and greater visual dependency.**

This observation motivates us to propose **S**patiotemporal **P**ath-**A**ware **R**easoning paradigm based on experiential **K**nowledge(SPARK). Under this paradigm, we encourage VLMs to autonomously and diversely explore spatiotemporal reasoning paths, and to acquire transferable experience from their quality discrepancies. Specifically, we first design a visual-based fine reward to encourage VLMs to favor paths with high-quality potential. Building on this reward, we adopt MCTS to enable autonomous exploration of spatiotemporal reasoning paths. To address the inherent path redundancy and search inefficiency of vanilla MCTS [33, 34, 35], we introduce submodular optimization for greedy search-space pruning, removing redundant branches while preserving reasoning diversity. Finally, we present a training-free experience construction mechanism: inspired by Direct Preference Optimization (DPO) [36], we convert contrastive pairs of reasoning paths with distinct quality into structured experience, which is dynamically injected into the VLM through similarity-based retrieval during inference.

Our method features four prominent advantages. *i)* **Performance Advancement (§5.1)**: Extensive experiments on four open-source models (Qwen3.5/3-vl, InternVL3.5, LLaVA-NeXT) and three spatiotemporal benchmarks (VLM4D [11], DSI [37], VSI [38]) show that SPARK adaptively captures task-aligned high-quality reasoning patterns, achieving substantial performance gains. For instance, it enables Qwen3.5-9B [39] to reach 61.17% average accuracy, outperforming the Gemini-3-Pro (57.20%) [40]. *ii)* **Enhanced Efficiency (§5.5)**: As a training-free framework, it matches SOTA baselines with only a small set of prior samples, significantly reducing data annotation and computational costs. *iii)* **High Flexibility (§5.3)**: Designed as a plug-and-play test-time module, the SPARK seamlessly integrates with post-training techniques like SFT and GRPO [41] without architectural modifications. *iv)* **Strong Generalization (§5.4)**: Structured textual experience from spatiotemporal reasoning paths generalizes effectively across different datasets and VLM models.

2 Related Work

Spatiotemporal Intelligence in VLMs. Spatiotemporal intelligence focuses on endowing artificial intelligence (AI) with human-like understanding capabilities regarding the physical world, which is widely regarded as a pivotal cornerstone for realizing General Embodied AI [42, 43]. Fueled by the strong reasoning capability of Large Language Models (LLMs) [44, 45, 46, 47, 48], the research field has undergone a paradigm shift toward unified VLMs [49, 50]. By leveraging cross-modal alignment, modern VLMs move beyond isolated modal modeling to achieve holistic, semantic-aware understanding of complex dynamic scenes [39, 51, 52]. Therefore, current spatiotemporal intelligence mainly evolves along two lines: large-scale expert distillation and RL [14, 19]. The former inherits

LLM instruction alignment, distilling geometric priors from SOTA domain experts into task-specific modules and learnable parameters of VLMs [15, 16]. For example, Fan et al.[13] incorporates 3D geometric priors distilled from reconstruction experts into its geometry encoder to support precise spatial reasoning in VLMs. In contrast, RL enables VLMs to develop spatiotemporal intelligence via autonomous exploration, and enhances reasoning by maximizing cumulative rewards [17, 18]. Nevertheless, both paradigms suffer from poor generalization and non-transferable parameterization, constrained by limited teacher knowledge and sparse reward signals. To tackle these limitations, our work converts autonomous exploration into explicit textual experiences, facilitating efficient capability enhancement and cross-framework knowledge generalization for VLMs.

Experiential Self-Evolution. Different from conventional training optimization, experience-driven self-evolution has recently emerged as a promising paradigm to empower models [23, 25, 26]. These methods enable models to evolve autonomously by extracting reusable empirical knowledge from historical reasoning paths, especially erroneous failure paths [29, 53, 24]. For instance, Wu et al.[54] distills historical trajectories offline into reusable policy principles, which are then retrieved during online interactions to guide decision-making. And Zhou et al.[55] enhances generalization in complex tasks by autonomously discovering reusable reasoning structures from historical reasoning. However, most existing experience-driven frameworks rely on outcome-based evaluation, which treats the final success or failure as the sole signal for experience extraction [30, 8, 31, 56]. Our work shifts this focus from binary outcomes to fine-grained process assessment, ensuring that the distilled experiences capture the nuances of spatiotemporal reasoning and visual information.

3 Visual Information Drives Spatiotemporal Reasoning

In this section, we examine the correlation between visual input and reasoning path from the perspective of anchoring and dependency for visual information, aiming to clarify: *What properties do high-quality reasoning paths exhibit for visual input?* Therefore, we introduce two metrics: Visual Anchoring (VA) and Visual Dependency (VD). Specifically, VA measures the concentration level of visual information contributions to reasoning path generation. A higher VA value indicates that the visual contribution is highly peaked at a specific few regions, reflecting a decisive anchoring on specific visual information. Conversely, a lower VA score suggests that the reasoning is less anchored in visual cues and may instead be driven by internal linguistic habits. VD quantifies the extent to which visual information contributes to the generation of a reasoning path. A higher VD score confirms that the reasoning path is observation-driven, where key logical transitions are triggered by visual cues rather than linguistic inertia.

Definition 4.1 (Visual Anchoring, VA). Given a video frame sequence $V = \{v_1, \ldots, v_{|V|}\}$, a query Q, let $r = \{w_1, \ldots, w_{|r|}\}$ denote the generated token sequence of a generated reasoning step, we first partition each video frame into sub-regions $v_i = \{x_i^1, \ldots, x_i^{|v_i|}\}$ [1]. We then quantify the contribution of each sub-region x_i^j to the generation of r by integrating its *Sufficiency* and *Necessity*:

$$C_i^j = \underbrace{\sum_{k=1}^{|r|} P_{\text{VLM}}(w_k \mid x_i^j, w_{<k}, Q)}_{\text{Sufficiency}} + \underbrace{\sum_{k=1}^{|r|} \Big(1 - P_{\text{VLM}}(w_k \mid V \setminus x_i^j, w_{<k}, Q)\Big)}_{\text{Necessity}}, \tag{1}$$

where C_i^j reflects the raw evidentiary weight of region x_i^j. To further capture the distributional characteristics of these grounding points, we derive the final VA score by measuring the sparsity of the normalized contribution scores $\bar{C}_i^j$ across all spatiotemporal regions:

$$\text{VA}(r) = 1 - \frac{-\sum_{i,j} \bar{C}_i^j \log(\bar{C}_i^j)}{\log(|V| \cdot |v_i|)}, \quad \text{where} \quad \bar{C}_i^j = \frac{(C_i^j)^\gamma}{\sum_{i,j}(C_i^j)^\gamma}. \tag{2}$$

Here, γ is a temperature coefficient that adjusts the sensitivity of the distribution.

[1] In this paper, we using SLICO [57] superpixel segmentation for this operation

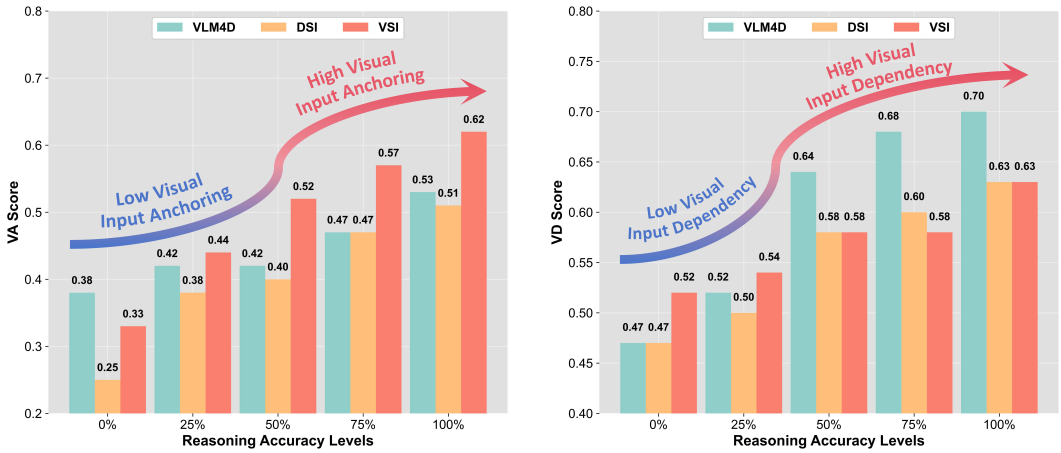


Figure 1: Reasoning path performance on VA (left) and VD (right) across accuracy levels.

Definition 4.2 (Visual Dependency, VD). For any token w_k in reasoning step r, we define the Visual Dependency Index I_k to quantify the VLM's reliance on visual input during generation:

$$I_k = \sum_{i,j} \left(P_{\text{VLM}}(w_k \mid x_{<i}^{<j}, w_{<k}, Q) - \min_{x_i^j \in V} P_{\text{VLM}}(w_k \mid x_i^j, w_{<k}, Q) \right). \tag{3}$$

A high I_k indicates that the generation of token w_k is highly sensitive to visual variations, signifying strong modality dependency. To eliminate biases arising from varying path lengths or filler language, we define the aggregate Visual Dependency (VD) score by averaging the top-5 most sensitive tokens:

$$\text{VD}(r) = \frac{1}{K} \sum_{k \in \mathcal{K}} I_k, \quad \text{where } \mathcal{K} = \text{Top-5}(I_k{}_{k=1}^{|r|}). \tag{4}$$

By focusing on these core visual-driven segments, $\text{VD}(r)$ provides a robust measure of reasoning faithfulness that remains invariant to response verbosity.

To explore the correlation between visual input and reasoning path quality, we adopt Qwen3.5-9B [39] for reasoning path collection on three spatio-temporal reasoning datasets: VLM4D Bench[11], DSI Bench[37], and VSI Bench[38]. We sample 200 samples from each dataset and leverage the standard MCTS method to explore reasoning paths for every sample. For each reasoning path, we employ the average accuracy of its results across four evaluations as the quality assessment metric [2]. We then categorize all reasoning paths into five accuracy levels: 0%, 25%, 50%, 75% and 100%, and compute the average VA and VD values for reasoning paths within each level.

As illustrated in Figure 1, reasoning paths of high accuracy appear two distinctive features. *i)* High-quality reasoning paths tend to focus on specific few pieces of sub-regions. For instance, Figure 1 (left) shows that the best-performed reasoning paths gain more than 0.6 of visual anchoring score in VSI bench. This indicates that a few regions contribute the majority of the information in the generation of high-quality inference paths. *ii)* High-quality reasoning paths show a pronounced reliance on visual information. As shown in Figure 1 (right) the best-performed reasoning paths have the most visual dependent score across all comparisons. This indicates that the generation of high-quality reasoning paths relies on the contribution of visual information. To summarize, high-quality reasoning paths are characterized by their selective focus on critical visual information and their strong dependence on visual information in the inference process. These observations highlight that effective visual reasoning relies on targeted utilization of key visual information, which inspires us to adjust the objective for experience extraction §4.1 optimization towards increasing VA and VD score. Appendix C.6 presents a case study to visually show VA and VD differences across reasoning paths of varied quality.

[2]We use the reasoning path as a starting point, inputting it into VLM as the prompt to guide model to continue generating until a final result is obtained, and then assess whether this result aligns with the ground truth. This operation is widely applied in outcome-based methods to assess the quality of reasoning paths [58, 59].

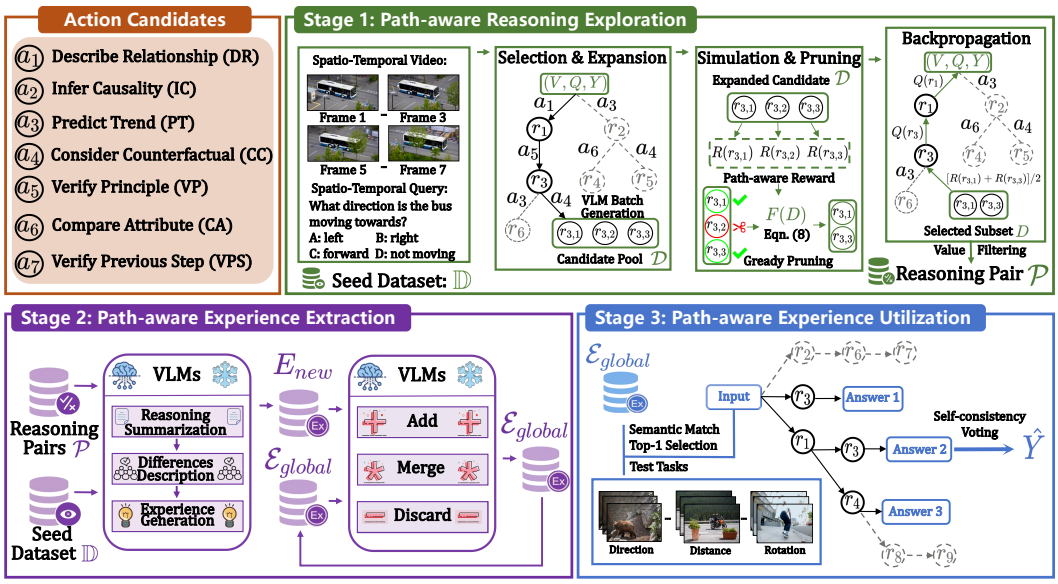


Figure 2: Overview of SPARK Framework. In path-aware reasoning exploration stage §4.1, VLMs adopt submodular optimized MCTS with visual reward for efficient exploration. In path-aware experience extraction stage §4.2, VLMs generate high-quality spatiotemporal reasoning experience based on the fine-grained differences between reasoning paths. In path-aware experience utilization stage §4.3, the adaptive retrieval experience guides VLMs to generate high-quality reasoning paths.

4 Methodology

Based on the findings presented in §3, we propose SPARK, designed to perceive the visual information in reasoning paths to overcome the inherent bottlenecks of traditional outcome-based experience extraction methods. As illustrated in Figure 2, our method contains a workflow for experience extraction that leverages an MCTS-based method to explore diverse reasoning paths and constructs structured experiences through pairs of distinct value paths.

4.1 Path-aware Reasoning Exploration

Formulation and Definition of Reasoning Tree Search. We formulate the exploration of each spatiotemporal sample from seed dataset $(V, Q, Y) \in \mathbb{D}$ as a tree search problem, where Y denotes the ground truth. The search state S_t is defined as the accumulated reasoning path $(V, Q, r_1, \ldots, r_t)$, where $S_0 = (V, Q)$ represents the root node. The subsequent node r_{t+1} denote one reasoning step generated by model $r_{t+1} = \text{VLM}(S_t; a_{t+1})$, where the subsequent action is sampled from action set $a_{t+1} \sim \mathcal{A}$. To effectively guide the VLM in simulating human-like cognitive decomposition, we first define a discrete action set $\mathcal{A}$ that atomizes complex spatiotemporal reasoning into manageable operations. Inspired by the reasoning atomization in OpenAI o1 [60] and related cognitive frameworks [61, 62], we introduce seven specific spatiotemporal reasoning actions: *Describe Relationship (DR, a1), Infer Causality (IC, a2), Predict Trend (PT, a3), Consider Counterfactual (CC, a4), Verify Principle (VP, a5), Compare Attributes (CA, a6), and Verify Previous Step (VPS, a7).* By sampling from this set $\mathcal{A}$, the model can explore the reasoning space in a structured and interpretable manner.

Monte Carlo Tree Search for Reasoning Exploration. We employ MCTS to iteratively navigate the reasoning paths. This design leverages MCTS to structured search over the reasoning space while prioritizing promising paths [8, 30]. The standard exploration follows a four-step cycle within each iteration. First, in the **Selection** phase, we navigate from the root node S_0 to a leaf node by applying the Upper Confidence Bounds applied to Trees (UCT) algorithm [63]. Second, during **Expansion**, for a selected leaf node r_t, the VLM executes an action $a_t \in \mathcal{A}$ to generate a candidate pool of reasoning steps $\mathcal{C} = \{r_{t+1,1}, \ldots, r_{t+1,n}\}$. Third, the **Simulation** phase evaluates the potential of each new node $r_{t+1,j} \in \mathcal{C}$ by conducting multiple rollouts until a terminal answer $\hat{Y}$ is reached. The outcome-based

node reward is typically estimated by the average accuracy: $R_{base}(r) = \frac{1}{K}\sum_{k=1}^{K} \mathbb{1}(\hat{Y} = Y)$, where K is the number of simulations. In **Backpropagation**, the simulated rewards are propagated upward to update the values $Q(r)$ and visit counts $N(r)$ along the selection path.

Path-aware Reward and Submodular Greedy Pruning. While vanilla MCTS supports systematic reasoning path exploration, it suffers notable limitations in spatiotemporal tasks. Its outcome-based reward fails to capture the fine-grained reasoning quality and unconstrained expansion of the candidate pool $\mathcal{C}$ brings path redundancy and low search efficiency [33, 34]. To bridge these gaps, we propose path-aware reward and submodular greedy Pruning strategy within the SPARK framework. Specifically, guided by our findings in §3, we define a composite reward $R(r_{t,j}) = R_{base}(r_{t,j}) + \alpha \cdot \text{VA}(r_{t,j}) + (1-\alpha) \cdot \text{VD}(r_{t,j})$ for each candidate $r_{t,j} \in \mathcal{C}$, where α is the balance weight, while VA and VD quantify the concentration and dependency of reasoning steps on visual input. This mechanism ensures that the exploration explicitly favors paths that faithfully rely on visual input. To optimize the search efficiency, we introduce a pruning step after simulation. We formulate the node selection as a submodular maximization problem:

$$F(D) = \sum_{r_{t,j} \in \mathcal{C}} \left(R(r_{t,j}) \cdot \max_{r_{t,k} \in D} \text{Sim}(r_{t,j}, r_{t,k}) \right), \quad \text{s.t. } |D| = m \leq |\mathcal{C}|, \tag{5}$$

where $\text{Sim}(\cdot)$ denotes the cosine similarity of VLM's hidden-layer representations. We formally prove in Appendix A.2 that $F(D)$ satisfies the properties of **monotonicity and submodularity**. Leveraging these properties, we employ the greedy pruning strategy: starting with $D = \emptyset$, we iteratively add the node $r^* = \arg\max_{r \in \mathcal{C} \setminus D}(F(D \cup \{r\}) - F(D))$ that yields the maximum marginal gain until the budget m is reached. This method theoretically guarantees a $(1 - 1/e)$ approximation of the optimal diversity-aware subset [64], effectively concentrating the search budget on low-redundant reasoning paths. Upon completing the search, we extract from the tree pairs of sibling nodes (r_{high}, r_{low}) whose value scores exceed a specified threshold λ, i.e., $Q(r_{high}) - Q(r_{low}) > \lambda$. These pairs are then aggregated into a dataset $\mathcal{P}$ to provide fine-grained supervision for distilling experiences. In Appendix A.3, We present the detailed exploration process by pseudocode and illustrations.

4.2 Path-aware Experience Extraction

Building upon the contrastive path pairs $\mathcal{P}$ collected in §4.1, we employ the VLM as a meta-optimizer to transform these raw paths into high-level, transferable spatiotemporal experiences. Specifically, for each spatiotemporal sample (V, Q, Y), the VLM acts as a cognitive judge by provided prompts to analyze the divergence between the corresponding reasoning paths $(r_{high}, r_{low}) \in \mathcal{P}$, distilling a localized reasoning experience $E_{new} = \text{VLM}(r_{high}, r_{low}, V, Q, Y)$. To prevent the redundant generation of identical experience, we formulate the construction of the sample-specific experience set $\mathcal{E}_{(V,Q,Y)}$ as a dynamic refinement process. Controlled by prompts, the VLM evaluates the semantic relationship between E_{new} and the existing experiences in $\mathcal{E}_{(V,Q,Y)}$ to dynamically execute one of three refinement operations:

Add: Append E_{new} as a novel experience if it shows low overlap with existing experiences.
Merge: Fuse E_{new} with similar experiences to form a more comprehensive reasoning rule.
Discard: Ignore E_{new} if it is identified as semantically redundant with existing experience.

Once the sample-specific experience sets $\mathcal{E}_{(V,Q,Y)}$ are constructed for all samples, they are integrated into a unified global experience set, formulated as $\mathcal{E}_{global} = \bigcup_{(V,Q,Y)} \mathcal{E}_{(V,Q,Y)}$, which serves as a high-quality prior knowledge set to guide the model in generating high-quality reasoning paths during subsequent spatiotemporal reasoning tasks. Appendix B.3 presents the specific prompts.

4.3 Path-aware Experience Utilization

In the inference phase, SPARK adaptively leverages the global experience bank $\mathcal{E}_{global}$ to facilitate the reasoning of test samples. Given a new input (V, Q), we first perform a semantic similarity search within $\mathcal{E}_{global}$ to retrieve the most relevant sample-specific set $\mathcal{E}_{(V^*,Q^*)}$:

$$(V^*, Q^*, Y^*) = \arg\max_{(V',Q')} \text{Sim}\left((V, Q), (V', Q')\right), \quad \text{s.t. } \mathcal{E}_{(V',Q',Y')} \in \mathcal{E}_{global}. \tag{6}$$

Subsequently, we independently prompt the VLM with each individual experience $E_i \in \mathcal{E}_{(V^*,Q^*,Y^*)}$ as a task-specific prompt. Finally, we employ a majority voting mechanism to aggregate the multiple

Models	VLM4D			DSI				VSI			Avg.
	Real	Syn.	ALL	Obj-Sc.	Obs-Sc.	Obs-Obj	ALL	Num.	MC	ALL	-
Proprietary Models											
GPT-4o [65]	60.03	49.92	57.55	40.32	31.39	46.98	37.23	31.57	36.13	33.79	42.86
GPT-5 [66]	64.48	63.21	64.17	38.06	35.95	60.99	40.02	49.65	44.93	47.36	50.52
Gemini2.5-Pro [67]	63.57	57.83	62.16	44.95	44.67	61.44	46.90	48.26	47.28	47.78	52.28
Gemini-3-Pro [40]	65.65	67.99	66.22	47.35	46.42	63.09	48.92	60.28	52.39	56.45	57.20
Open-Source Models											
Qwen2.5-VL-7B [49]	43.32	45.64	43.89	34.33	29.68	48.14	34.02	30.13	32.69	31.37	36.43
Qwen2.5-VL-32B [49]	47.13	51.03	48.09	37.45	32.70	48.32	36.77	31.47	34.65	33.02	39.29
Qwen3-VL-8B [50]	57.55	63.37	58.98	39.47	39.95	53.68	41.47	46.92	42.86	42.15	47.53
Qwen3-VL-32B [50]	58.30	65.28	60.01	41.68	41.28	56.73	43.40	50.43	48.53	49.51	50.97
Qwen3.5-9B [39]	65.28	67.87	65.91	43.76	41.03	57.62	44.30	56.37	53.28	54.87	55.03
Qwen3.5-27B [39]	65.73	68.93	66.51	43.68	42.32	58.83	44.99	59.32	58.70	59.02	56.84
InternVideo2.5-8B [68]	52.56	44.58	50.60	38.73	41.61	55.84	42.15	38.22	41.25	39.69	44.15
InternVL3.5-8B [51]	42.74	40.67	42.23	38.40	30.71	49.56	36.42	32.72	37.52	35.05	37.90
InternVL3.5-38B [51]	43.68	42.75	43.45	38.69	34.66	56.06	39.11	38.73	42.66	40.64	41.07
LLaVA-NeXT-7B [52]	38.25	29.97	36.22	34.68	29.49	42.00	33.32	30.78	29.34	30.08	33.21
LLaVA-One-Vision-7B [69]	36.04	37.58	36.42	32.35	27.43	43.47	31.59	29.52	32.19	30.82	32.94
3D Spatial Reasoning Models											
VLM-3R [13]	36.97	24.73	33.97	33.57	40.87	49.13	38.74	60.03	59.40	59.72	44.14
VG-LLM [14]	49.55	37.39	46.57	37.70	32.13	54.41	37.35	52.36	42.08	47.36	43.76
Spatial-MLLM [17]	39.85	33.92	38.40	33.59	28.84	52.33	41.99	52.48	42.07	47.42	42.60
Our Models											
SPARK (Qwen3.5-9B)	**71.33**	**71.46**	**71.36**	**50.54**	**48.81**	**64.13**	**51.49**	**60.54**	**60.77**	**60.65**	**61.17**
SPARK (Qwen3-VL-8B)	62.65	67.87	63.93	44.03	42.13	58.72	45.04	53.15	54.73	53.91	54.29
SPARK (InternVL3.5-8B)	47.39	46.27	47.12	40.82	42.53	57.24	43.64	37.19	41.63	39.35	43.37
SPARK (LLaVA-NeXT-7B)	42.50	34.83	40.62	38.41	33.44	46.25	37.21	35.87	34.00	34.96	37.60

Table 1: Comparison of the accuracy across VLM4D Bench, DSI Bench, and VSI Bench. Bold font indicates the best performance [3].

generated reasoning outcomes:

$$\hat{Y} = \arg\max_{Y \in \mathcal{Y}} \sum_{E_i \in \mathcal{E}_{(V^*, Q^*)}} \mathbb{1}\left(\text{VLM}(V, Q, E_i) = Y\right), \tag{7}$$

where $\mathbb{1}(\cdot)$ is the indicator function. This aggregation process effectively mitigates the bias of individual reasoning paths, synthesizing the collective wisdom of retrieved experiences into a high-fidelity final answer.

5 Experiments

To evaluate the effectiveness of SPARK, we conduct extensive experiments using Qwen3-VL-8B [50], Qwen3.5-9B [39], InternVL3.5-8B [51], and LLaVA-NeXT-7B [52] as backbones across three spatiotemporal reasoning benchmarks: VLM4D [11], DSI [37], and VSI [38]. We benchmark SPARK against 18 baseline models across three categories. Detailed information regarding the datasets, baselines and implementation specifics is provided in the Appendix B. We also analyze the superiority of SPARK from various aspects: §5.2 Ablation Study, §5.3 Flexibility, §5.4 Cross-model Generalization, §5.5 Data Scaling, §C.1 Cross-Domain Generalization, §C.2 Inference Time Scaling, §C.3 Comparison with Other Search Based Methods, §C.4 Computational Cost,§C.5 Statistical Saliency.

5.1 Performance Comparison

We report the experimental results in Table 1. We find that SPARK advances the state-of-the-art in spatio-temporal reasoning task, with its Qwen3.5-9B variant achieving a leading average accuracy of 61.17%. This performance not only dominates the open-source field but also exceeds proprietary models such as GPT-5 (50.52%) and Gemini-3-Pro (57.20%), particularly on VLM4D where it hits a peak of 71.36%. These gains are remarkably consistent across various backbones, yielding

[3]The ALL means overall accuracy. For VLM4D, we show 2 categories: Real and Syn. (synthetic). For DSI, we present 3 categories: Obj-Sc. (Object-Scene), Obs-Sc. (Obs-Sc.), Obs-Obj (Observer-Object). For VSI, we present 2 categories: Num. (Numerical Answer) and MC (Multiple-Choice Answer).

improvements of 6.14% and 6.76% for the Qwen3.5 and Qwen3-vl series and maintaining steady growth even on base models like LLaVA-NeXT and InternVL3.5. Beyond general accuracy, SPARK demonstrates superior stability compared to dedicated 3D reasoning models. While specialized methods like VLM-3R can reach performance levels comparable to ours on specific benchmarks such as VSI-Bench, their efficacy often fluctuates across more diverse scenarios. In contrast, SPARK maintains consistent and robust improvements across the entire evaluation suite. By adaptively retrieving high-quality reasoning experiences, our framework effectively bridges the gap between narrow, task-specific perception and generalized, expert-level spatiotemporal understanding.

5.2 Ablation Study

To comprehensively investigate the contribution of each component within SPARK to its final performance, we conduct an extensive ablation study using Qwen3.5-9B as the representative backbone. Specifically, we evaluate the following variants: (1) w/o VD: The VD metric is removed from the reward function in §4.1. (2) w/o VA: The VA is excluded from the reward function in §4.1. (3) w/o Pruning: The submodular greedy pruning strategy is removed, reverting the exploration process to a standard MCTS algorithm. (4) w/ PT: The synthesized experiences are replaced with raw reasoning pairs as prompts for the VLM. (5) w/ SC: The adaptive experience retrieval mechanism is substituted with a standard self-consistency decoding method. The results in Table 2 confirm that SPARK achieves the best performance, underscoring the necessity of each integrated module. Notably, the performance degradation in the w/o VD (-3.38%) and w/o VA (-2.41%) variants demonstrates that pilot-study-guided rewards successfully steer the model toward more visually-grounded and factually accurate reasoning paths. Moreover, the significant decline observed in the w/o Pruning variant (-4.54%) highlights the efficiency of our exploration strategy which proves that submodular pruning is essential for navigating the search space efficiently.

Model Types	VLM4D	DSI	VSI	Avg.
SPARK	**71.36**	**51.49**	**60.65**	**61.17**
- w SC	69.20	48.73	57.38	58.44$_{2.73\downarrow}$
- w PT	66.82	47.31	54.53	56.22$_{4.95\downarrow}$
- w/o VD	69.47	48.57	58.33	57.79$_{3.38\downarrow}$
- w/o VA	68.10	50.55	57.62	58.76$_{2.41\downarrow}$
- w/o Pruning	66.23	45.95	57.70	56.63$_{4.54\downarrow}$

Table 2: Ablation Study on SPARK (Qwen3.5-9B).We use color ■ to indicate the performance changes of different variants relative to the original method.

5.3 Analysis on Flexibility

While Table 1 demonstrates SPARK's flexibility across various VLM backbones, we further investigate its compatibility with models derived from state-of-the-art post-training paradigms. Specifically, we apply SPARK to Spatial-MLLM [17], an RL-enhanced model trained via Group Relative Policy Optimization (GRPO) [41] based on Qwen2.5-VL-3B [49]. As illustrated in Figure 3, SPARK consistently yields performance gains when integrated with this post-trained reasoning model across all benchmarks. For instance, SPARK achieves 42.83% on VLM4D-Bench, a significant 4.43% improvement over the strong RL-tuned baseline (38.40%). These results position SPARK as a enhancement that can be seamlessly layered onto advanced reasoning-enhanced VLMs to break through existing performance plateaus.

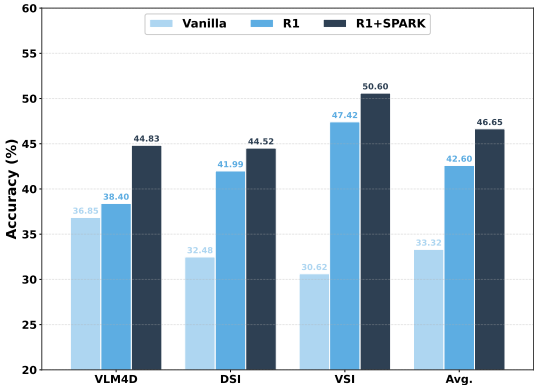


Figure 3: Analysis on SPARK flexibility. We evaluate three model variants: (i) Vanilla: original Qwen2.5-VL-3B [49]. (ii) R1: reasoning-enhanced Spatial-MLLM [17] based on the vanilla model. (iii) R1+SPARK: R1 integrated with SPARK.

5.4 Analysis on Cross-model Generalization

In this section, we investigate whether the reasoning experiences curated by SPARK possess the capacity for cross-model generalization. Specifically, we apply the experience set generated by SPARK (using Qwen3.5-9B [39] as the source) to significantly larger target backbones: Gemini-3-Pro

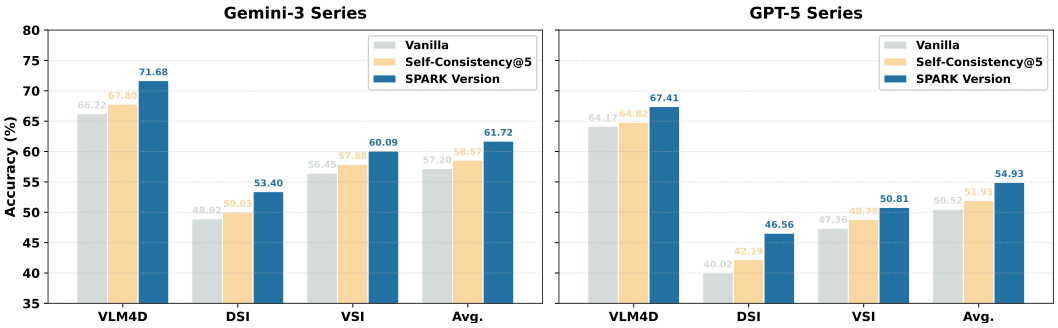


Figure 4: Analysis on Generalization. Evaluations on Gemini-3-Pro and GPT-5 confirm the high generalization of SPARK's reasoning experiences.

[40] and GPT-5 [66]. As illustrated in Figure 4, the experimental results underscore the exceptional transferability of SPARK, which consistently yields performance improvements across all evaluated benchmarks. Notably, Gemini-3-Pro and GPT-5 achieve average accuracy gains of 4.52% and 4.41%, respectively, also outperforming the standard test-time scaling baseline, Self-Consistency@5. These findings suggest that SPARK effectively distills universal, model-agnostic spatiotemporal reasoning patterns rather than merely overfitting to its source architecture. Furthermore, this success validates a promising small-to-large enhancement paradigm: specialized compact models can curate high-quality reasoning insights that effectively catalyze the cognitive performance of frontier large-scale models.

5.5 Analysis on Data Scaling

In this section, we investigate the influence of seed dataset scale on the ultimate performance of SPARK. Using Qwen3.5-9B [39] as the representative backbone, we conducted experiments across five distinct seed dataset sizes, with results illustrated in Figure 5. Overall, SPARK exhibits strong scalability, as evidenced by a consistent performance upward trend: the average accuracy improves from 66.32% (with 50 samples) to 72.50% (with 1,000 samples) in VLM4D benchmark. Notably, the high data efficiency of SPARK is evident in its ability to deliver substantial performance gains with as few as 200 samples, making it a highly viable solution for resource-limited scenarios. Consequently, we adopt a default seed size of 200 samples, a strategic choice that yields an ideal balance between superior reasoning accuracy and the associated computational costs.

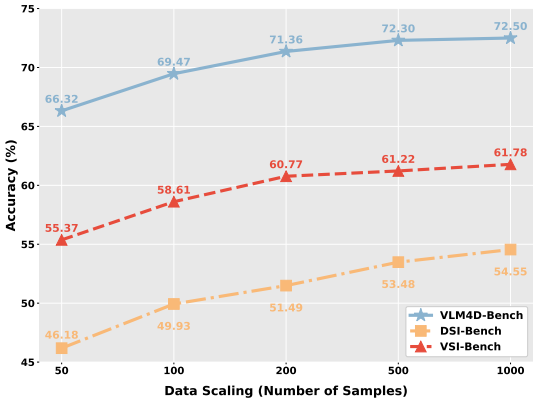


Figure 5: Analysis on Data Scaling.

5.6 Reasoning Paths Redundancy Reduction

To mitigate the path redundancy inherent in standard MCTS exploration [33, 34, 35], we introduced a submodular-based greedy pruning strategy in §4. In this section, we evaluate the effectiveness of SPARK in suppressing such redundancy by measuring the semantic overlap between generated reasoning paths. Specifically, we calculate the average semantic similarity between paths by measuring the cosine similarity of their hidden-layer representations, which are extracted from the Qwen3.5-9B backbone. We report the final metric

Model	VLM4D	DSI	VSI	Average
Mulberry[70]	0.48	0.53	0.53	0.51
LlamaV-o1[62]	0.48	0.42	0.46	0.45
LLaVA-CoT[59]	0.47	0.45	0.42	0.45
AR-MCTS[71]	0.52	**0.60**	0.58	0.56
Astar[8]	0.48	0.53	0.56	0.52
SPARK	**0.54**	0.59	**0.61**	**0.58**

Table 3: Redundancy comparison with search-based methods using Qwen3.5-9B as backbone.

as $(1 -$ Average Semantic Similarity), where a higher value signifies lower redundancy and greater exploratory diversity. As shown in Table 3, SPARK consistently achieves the highest scores across most benchmarks, reaching an overall average of 0.58. This outperforms other exploration-based baselines such as Mulberry (0.51), LlamaV-o1 (0.45), and Astar (0.52). These results empirically validate that our submodular pruning strategy effectively filters out semantically repetitive branches, ensuring that the exploration budget is concentrated on distinct and informative reasoning paths.

5.7 Case Study

In this section, we present a case study using Qwen3.5-9B across the VLM4D, DSI, and VSI datasets to analyze the reasoning behavior. As illustrated in Figure 8 to Figure 13, we compare the reasoning paths of the same input across different accuracy levels. Specifically, the heatmaps overlaid on the visual frames represent the attribution C_i^j in Eqn. (1), denoting the contribution of different sub-regions to the reasoning path. The background color of the reasoning tokens below corresponds to the VD metric in Eqn. (4) for each token. Experimental results demonstrate that, for high-quality paths, the sub-regions exhibiting higher contributions C_i^j are precisely concentrated on task-relevant objects within the visual input, thereby implying higher VA scores. Meanwhile, more tokens in high-quality paths show a high VD score on visual input. Conversely, low-quality reasoning paths exhibit dispersed contributions from sub-regions and a reliance on linguistic habits.

6 Conclusion

In this work, we explore the potential of experiential self-evolution to enhance the spatiotemporal reasoning of Vision-Language Models (VLMs). We observe that high-quality reasoning paths exhibit distinct characteristics of stronger visual anchoring (VA) and greater visual dependency (VD) compared to lower-quality trajectories. Inspired by this discovery, we propose a **S**patiotemporal **P**ath-**A**ware **R**easoning paradigm based on experiential **K**nowledge, namely SPARK. This method integrates MCTS with fine-grained visual rewards and submodular greedy pruning to enable the autonomous and efficient exploration of diverse reasoning paths. To facilitate model evolution, we design a training-free mechanism that distills contrastive path pairs into structured, transferable experiences, which are then adaptively retrieved to guide the VLM during inference. Extensive experiments demonstrate that SPARK consistently outperforms existing baselines across multiple benchmarks. Further analysis verifies that the distilled experiences yield strong cross-model generalization, providing an effective paradigm for spatiotemporal intelligence enhancement.

7 Limitations

Despite the effectiveness of SPARK, this study has several limitations. First, our pilot experiments only covered a limited variety of dataset and vlms. In future studies, we plan to extend to more diverse of datasets and vlms. Second, compared to standard MCTS, the integration of visual reward modeling introduces additional computational cost during the exploration phase. We will explore more efficient variants to mitigate this overhead in future work. Finally, due to limitations in computational resources, we have not yet conducted experiments on larger models (e.g., 70B+ parameters). In the future, should conditions permit, we will further explore the application of SPARK in larger-scale models.

8 Acknowledgements

This work is supported by National Key R&D Program (2023YFB3107400), National Natural Science Foundation of China (62272371, 62103323, 62521002, U24B20185, U2441240, 62132011), Initiative Postdocs Supporting Program (BX20190275, BX20200270), China Postdoctoral Science Foundation (2019M663723, 2021M692565), Natural Science Foundation of Shaanxi(2026JC-YXQN-197) and Fundamental Research Funds for the Central Universities under grant (xzy012024144, xzy012025043). Cong Wang was supported in part by the Research Grants Council of HK SAR under Grants CityU 11219025, 11219524, R1012-21, RFS2122-1S04, C6015-23G, and CRS-HKUST601/24. This work was also supported in part by the InnoHK initiative of the Government of the HKSAR and the Laboratory for AI-Powered Financial Technologies (AIFT), and the Guangdong and Hong Kong Universities "1+1+1" Joint Research Collaboration Scheme. Thanks to the New Cornerstone Science Foundation, the Xplorer Prize and K. C. Wong Education Foundation.

References

[1] Kazuaki Akamatsu, Tomohiro Nishino, and Yoichi Miyawaki. Spatiotemporal bias of the human gaze toward hierarchical visual features during natural scene viewing. *Scientific Reports*, 13(1):8104, 2023.

[2] Michail Maniadakis and Panos Trahanias. Temporal cognition: a key ingredient of intelligent systems. *Frontiers in neurorobotics*, 5:2, 2011.

[3] Ruishan Guo, Ciyu Ruan, Haoyang Wang, Zihang Gong, Jingao Xu, and Xinlei Chen. x^2-fusion: Cross-modality and cross-dimension flow estimation in event edge space. *arXiv preprint arXiv:2603.16671*, 2026.

[4] Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Xiang An, Yan Feng, Peng Pei, Xunliang Cai, et al. Think with 3d: Geometric imagination grounded spatial reasoning from limited views. *arXiv preprint arXiv:2510.18632*, 2025.

[5] Yuzhi Huang, Jie Wu, Weijue Bu, Ziyi Xiong, Gaoyang Jiang, Ye Li, Kangye Ji, Shuzhao Xie, Yue Huang, Chenglei Wu, et al. Robostream: Weaving spatio-temporal reasoning with memory in vision-language models for robotics. *arXiv preprint arXiv:2603.12939*, 2026.

[6] Nimesha Ranasinghe, Pravar Jain, Shienny Karwita, David Tolley, and Ellen Yi-Luen Do. Ambiotherm: enhancing sense of presence in virtual reality by simulating real-world environmental conditions. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 1731–1742, 2017.

[7] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018.

[8] Jinyang Wu, Mingkuan Feng, Guocheng Zhai, Shuai Zhang, Zheng Lian, Fangrui Lv, Pengpeng Shao, Ruihan Jin, Zhengqi Wen, and Jianhua Tao. Astar: Boosting multimodal reasoning with automated structured thinking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33926–33934, 2026.

[9] David Venuto, Sami Nur Islam, Martin Klissarov, Doina Precup, Sherry Yang, and Ankit Anand. Code as reward: empowering reinforcement learning with vlms. In *Proceedings of the 41st International Conference on Machine Learning*, pages 49368–49387, 2024.

[10] Alex Jinpeng Wang, Pan Zhou, Mike Zheng Shou, and Shuicheng Yan. Enhancing visual grounding in vision-language pre-training with position-guided text prompts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3406–3421, 2023.

[11] Shijie Zhou, Alexander Vilesov, Xuehai He, Ziyu Wan, Shuwang Zhang, Aditya Nagachandra, Di Chang, Dongdong Chen, Xin Eric Wang, and Achuta Kadambi. Vlm4d: Towards spatiotemporal awareness in vision language models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8600–8612, 2025.

[12] Azmine Toushik Wasi, Shahriyar Zaman Ridoy, Koushik Ahamed Tonmoy, Kinga Tshering, SM Hasan, Wahid Faisal, Tasnim Mohiuddin, and Md Rizwan Parvez. Timespot: Benchmarking geo-temporal understanding in vision-language models in real-world settings. *arXiv preprint arXiv:2603.06687*, 2026.

[13] Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, et al. Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction. *arXiv preprint arXiv:2505.20279*, 2025.

[14] Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors. *arXiv preprint arXiv:2505.24625*, 2025.

[15] Mahir Labib Dihan, MD Tanvir Hassan, MD TANVIR PARVEZ, Md Hasebul Hasan, Md Almash Alam, Muhammad Aamir Cheema, Mohammed Eunus Ali, and Md Rizwan Parvez. Mapeval: A map-based evaluation of geo-spatial reasoning in foundation models. In *International Conference on Machine Learning*, 2025.

[16] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. Mmsi-bench: A benchmark for multi-image spatial intelligence. *arXiv preprint arXiv:2505.23764*, 2025.

[17] Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. *arXiv preprint arXiv:2505.23747*, 2025.

[18] Xingyilang Yin, Chengzhengxu Li, Jiahao Chang, Chi-Man Pun, and Xiaodong Cun. Mllm-4d: Towards visual-based spatial-temporal intelligence. *arXiv preprint arXiv:2603.00515*, 2026.

[19] Ting Huang, Zeyu Zhang, and Hao Tang. 3d-r1: Enhancing reasoning in 3d vlms for unified scene understanding. *arXiv preprint arXiv:2507.23478*, 2025.

[20] Sunbowen Lee, Junting Zhou, Chang Ao, Kaige Li, Xeron Du, Sirui He, Haihong Wu, Tianci Liu, Jiaheng Liu, Hamid Alinejad-Rokny, et al. Quantification of large language model distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4985–5004, 2025.

[21] Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

[22] Sicheng Feng, Kaiwen Tuo, Song Wang, Lingdong Kong, Jianke Zhu, and Huan Wang. Rewardmap: Tackling sparse rewards in fine-grained visual reasoning via multi-stage reinforcement learning. *arXiv preprint arXiv:2510.02240*, 2025.

[23] Guangxiang Zhao, Qilong Shi, Xusen Xiao, Xiangzheng Zhang, Tong Yang, and Lin Sun. Thinking with reasoning skills: Fewer tokens, more accuracy. *arXiv preprint arXiv:2604.21764*, 2026.

[24] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642, 2024.

[25] John Chen, Xi Lu, Yuzhou Du, Michael Rejtig, Ruth Bagley, Mike Horn, and Uri Wilensky. Learning agent-based modeling with llm companions: Experiences of novices and experts using chatgpt & netlogo chat. In *Proceedings of the 2024 CHI conference on human factors in computing systems*, pages 1–18, 2024.

[26] Yanyun Wang, Xumei Fang, Zan Xu, Jianye Li, and Luping Wang. Exploring the impact of explainability in large language model (llm) applications on user experience. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–8, 2025.

[27] Marc-Antoine Allard, Arnaud Teinturier, Victor Xing, and Gautier Viaud. Experiential reflective learning for self-improving llm agents. *arXiv preprint arXiv:2603.24639*, 2026.

[28] Xuanbo Su, Yingfang Zhang, Hao Luo, Xiaoteng Liu, and Leo Huang. Mistake notebook learning: Selective batch-wise context optimization for in-context learning. *arXiv preprint arXiv:2512.11485*, 2025.

[29] Kai Zhang, Xiangchao Chen, Bo Liu, Tianci Xue, Zeyi Liao, Zhihan Liu, Xiyao Wang, Yuting Ning, Zhaorun Chen, Xiaohan Fu, et al. Agent learning via early experience. *arXiv preprint arXiv:2510.08558*, 2025.

[30] Shuhang Liu, Zhenrong Zhang, Pengfei Hu, Jiefeng Ma, Jun Du, Qing Wang, Jianshu Zhang, Quan Liu, Jianqing Gao, and Feng Ma. Mmc: Iterative refinement of vlm reasoning via mcts-based multimodal critique. In *Proceedings of the 3rd International Workshop on Large Generative Models Meet Multimodal Applications*, pages 11–20, 2025.

[31] Yikun Wang, Siyin Wang, Qinyuan Cheng, Zhaoye Fei, Liang Ding, Qipeng Guo, Dacheng Tao, and Xipeng Qiu. Visuothink: Empowering lvlm reasoning with multimodal tree search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21707–21719, 2025.

[32] Haobo Yuan, Yueyi Sun, Yanwei Li, Tao Zhang, Xueqing Deng, Henghui Ding, Lu Qi, Anran Wang, Xiangtai Li, and Ming-Hsuan Yang. Visual reasoning tracer: Object-level grounded reasoning benchmark. *arXiv preprint arXiv:2512.05091*, 2025.

[33] Liam Schramm and Abdeslam Boularias. Provably efficient long-horizon exploration in monte carlo tree search through state occupancy regularization. In *International Conference on Machine Learning*, pages 43828–43861. PMLR, 2024.

[34] Steven James, George Konidaris, and Benjamin Rosman. An analysis of monte carlo tree search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[35] Rui Ha, Chaozhuo Li, Rui Pu, Litian Zhang, Xi Zhang, and Sen Su. Dsg-mcts: A dynamic strategy-guided monte carlo tree search for diversified reasoning in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10541–10555, 2025.

[36] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

[37] Ziang Zhang, Zehan Wang, Guanghao Zhang, Weilong Dai, Yan Xia, Ziang Yan, Minjie Hong, and Zhou Zhao. Dsi-bench: A benchmark for dynamic spatial intelligence. *arXiv preprint arXiv:2510.18873*, 2025.

[38] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10632–10643, 2025.

[39] Qwen Team. Qwen3. 5-omni technical report. *arXiv preprint arXiv:2604.15804*, 2026.

[40] Google DeepMind. Gemini 3. `https://deepmind.google/models/gemini/`, 2025. Accessed: 2025-11-01.

[41] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[42] Yann LeCun et al. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[43] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019.

[44] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[45] Yukang Yang, Declan Campbell, Kaixuan Huang, Mengdi Wang, Jonathan Cohen, and Taylor Webb. Emergent symbolic mechanisms support abstract reasoning in large language models. *arXiv preprint arXiv:2502.20332*, 2025.

[46] Ruishan Guo, Yibing Liu, Guoxin Ma, Yan Wang, Yueyang Zhang, Long Xia, Kecheng Chen, Zhiyuan Sun, and Daiting Shi. When less is more: The llm scaling paradox in context compression. *arXiv preprint arXiv:2602.09789*, 2026.

[47] Chengzhengxu Li, Xiaoming Liu, Zhaohan Zhang, Shengchao Liu, Guoxin Ma, Yu Lan, Cong Wang, and Chao Shen. Can reasoning path still be effective as input? bridging post-reasoning to chain-of-thought compression. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 39014–39034, 2026.

[48] Zhaohan Zhang, Junjie Liu, Chengzhengxu Li, Chen Shen, Xiaoming Liu, Chao Shen, Jieping Ye, Ziquan Liu, and Ioannis Patras. When confidence rises too early: Detecting shortcut reasoning via premature answer commitment. *arXiv preprint arXiv:2609.35074*, 2026.

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

[50] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[51] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[52] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024.

[53] XIANGYU PENG, Congying Xia, Xinyi Yang, Caiming Xiong, Chien-Sheng Wu, and Chen Xing. Regenesis: Llms can grow into reasoning generalists via self-improvement. In *The Thirteenth International Conference on Learning Representations*.

[54] Rong Wu, Xiaoman Wang, Jianbiao Mei, Pinlong Cai, Daocheng Fu, Cheng Yang, Licheng Wen, Xuemeng Yang, Yufan Shen, Yuxin Wang, et al. Evolver: Self-evolving llm agents through an experience-driven lifecycle. *arXiv preprint arXiv:2510.16079*, 2025.

[55] Pei Zhou, Jay Pujara, Xiang Ren, Xinyun Chen, Heng-Tze Cheng, Quoc V Le, Denny Zhou, Swaroop Mishra, Huaixiu S Zheng, et al. Self-discover: Large language models self-compose reasoning structures. *Advances in Neural Information Processing Systems*, 37:126032–126058, 2024.

[56] Chengzhengxu Li, Xiaoming Liu, Yichen Wang, Duyi Li, Yu Lan, and Chao Shen. Dialogue for prompting: A policy-gradient-based discrete prompt generation for few-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18481–18489, 2024.

[57] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

[58] Yuda Song, Julia Kempe, and Rémi Munos. Outcome-based exploration for llm reasoning. In *NeurIPS 2025 Workshop: Second Workshop on Aligning Reinforcement Learning Experimentalists and Theorists*.

[59] Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. Llava-cot: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2087–2098, 2025.

[60] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[61] Ling Yang, Zhaochen Yu, Tianjun Zhang, Shiyi Cao, Minkai Xu, Wentao Zhang, Joseph E Gonzalez, and Bin Cui. Buffer of thoughts: Thought-augmented reasoning with large language models. *Advances in Neural Information Processing Systems*, 37:113519–113544, 2024.

[62] Omkar Thawakar, Dinura Dissanayake, Ketan Pravin More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan, Yuhao Li, Ilmuz Zaman Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-o1: Rethinking step-by-step visual reasoning in llms. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24290–24315, 2025.

[63] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

[64] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

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

[66] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

[67] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[68] Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025.

[69] Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, et al. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025.

[70] Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, Yuxin Song, Haocheng Feng, Li Shen, et al. Mulberry: Empowering mllm with o1-like reasoning and reflection via collective monte carlo tree search. *arXiv preprint arXiv:2412.18319*, 2024.

[71] Guanting Dong, Chenghao Zhang, Mengjie Deng, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. Progressive multimodal reasoning via active retrieval. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3579–3602, 2025.

[72] Chen Yin and Zixuan Zhang. A study of sentence similarity based on the all-minilm-l6-v2 model with "same semantics, different structure" after fine tuning. In *2024 2nd International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2024)*, pages 677–684. Atlantis Press, 2024.

[73] Samuel Kotz and Norman L Johnson. *Breakthroughs in statistics: methodology and distribution*. Springer Science & Business Media, 2012.

Appendix of SPARK

This comprehensive supplementary material provides in-depth insights into SPARK, covering preliminaries, experimental details, and additional results. The appendix is organized as follows:

Contents

A Preliminaries 17
- A.1 Task Definition 17
- A.3 Theoretical Analysis of the Pruning Objective 17
- A.2 Detailed Algorithmic Workflow 18

B Experimental Details 18
- B.1 Datasets 19
- B.2 Baselines 19
- B.3 Implementation Details 20

C Additional Results 21
- C.1 Analysis on Cross-Domain Generalization 21
- C.2 Analysis on Inference Time Scaling 22
- C.3 Comparison with Other Search Based Methods 22
- C.4 Analysis on Computational Cost 23
- C.5 Analysis on Statistical Saliency 23
- C.6 Case Study 24

A Preliminaries

In this section, we summarize the definitions of spatiotemporal reasoning tasks §A.1, and present the preliminary Monte Carlo Tree Search (MCTS) mechanism adopted in our method §A.3. In addition, we provide the detailed theoretical proof of the monotonicity and submodularity property for our pruning objective function §A.2, to further validate the rationality of our proposed experience construction strategy.

A.1 Task Definition

This section exhibits the definitions of the spatiotemporal reasoning tasks.

Spatiotemporal Reasoning. We formalize the spatiotemporal understanding task as a state-space search problem with a quadruple $(\mathcal{V}, \mathcal{Q}, \mathcal{R}, \mathcal{Y})$, where $\mathcal{V}$ denotes the video input space, $\mathcal{Q}$ the natural language query space, $\mathcal{R}$ the intermediate reasoning step space, and $\mathcal{Y}$ the final answer space. Given a video frame sequence $V = \{v_1, \ldots, v_{|V|}\} \in \mathcal{V}$ and a query $Q \in \mathcal{Q}$, the VLM iteratively generates a reasoning path $R = \{r_1, \ldots, r_{|R|}\} \in \mathcal{R}$:

$$r_i = \text{VLM}(r_{<i}; V, Q), \quad 1 \leq i \leq |R|, \tag{8}$$

where r_i represents an individual reasoning step. Subsequently, the VLM leverages the context (V, Q, R) to output the final prediction $\hat{Y} \in \mathcal{Y}$:

$$\hat{Y} = \text{VLM}(R, V, Q). \tag{9}$$

The objective of spatiotemporal reasoning is to enable the VLMs to derive the optimal reasoning path R^*, maximizing $P_{\text{VLM}}(\cdot)$ for predicting ground truth answers $Y \in \mathcal{Y}$:

$$R^* = \arg\max_{R \in \mathcal{R}} P_{\text{VLM}}(\hat{Y} = Y \mid V, Q, R). \tag{10}$$

A.2 Theoretical Analysis of the Pruning Objective

In this section, we provide the theoretical justification for employing a greedy algorithm to optimize our pruning objective. Finding the exact optimal subset $D \subseteq \mathcal{D}$ of size K that maximizes the objective function is computationally intractable (NP-Hard). However, the greedy strategy can provide a strong $(1 - 1/e)$ approximation guarantee if the objective function is both *monotonic increasing* and *submodular* [64]. Simply, our objective function can be defined as:

$$F(D) = \sum_{x \in \mathcal{D}} \left(R(x) \cdot \max_{y \in D} Sim(x, y) \right), \tag{11}$$

where $\mathcal{D}$ is the ground set of all candidate paths, $D \subseteq \mathcal{D}$ is the selected subset, $R(x) \geq 0$ denotes the non-negative reward, and $Sim(x, y)$ denotes the similarity metric.

Theorem 1 (Monotonicity). The function $F(D)$ is monotonic increasing. That is, for any subsets $A \subseteq B \subseteq \mathcal{D}$, we have $F(A) \leq F(B)$.

Proof.

For any element $x \in \mathcal{D}$, since $A \subseteq B$, the maximum similarity score over set B must be greater than or equal to the maximum similarity score over set A:

$$\max_{y \in A} Sim(x, y) \leq \max_{y \in B} Sim(x, y). \tag{12}$$

Given that the reward function yields non-negative values ($R(x) \geq 0$), multiplying both sides by $R(x)$ preserves the inequality:

$$R(x) \cdot \max_{y \in A} Sim(x, y) \leq R(x) \cdot \max_{y \in B} Sim(x, y). \tag{13}$$

Summing this inequality over all elements $x \in \mathcal{D}$ yields:

$$\sum_{x \in \mathcal{D}} \left(R(x) \cdot \max_{y \in A} Sim(x, y) \right) \leq \sum_{x \in \mathcal{D}} \left(R(x) \cdot \max_{y \in B} Sim(x, y) \right). \tag{14}$$

Thus, $F(A) \leq F(B)$, which concludes the proof of monotonicity. ■

Theorem 2 (Submodularity). The function $F(D)$ is submodular. That is, for any subsets $A \subseteq B \subseteq \mathcal{D}$ and any element $e \in \mathcal{D} \setminus B$, the marginal gain satisfies $F(A \cup \{e\}) - F(A) \geq F(B \cup \{e\}) - F(B)$.

Proof.

Let us denote the marginal gain of adding element e to a set S as $\Delta F(e \mid S) = F(S \cup \{e\}) - F(S)$. By substituting the definition of $F(S)$, the marginal gain can be written as:

$$\Delta F(e \mid S) = \sum_{x \in \mathcal{D}} R(x) \left[\max_{y \in S \cup \{e\}} Sim(x, y) - \max_{y \in S} Sim(x, y) \right]. \tag{15}$$

Notice that $\max_{y \in S \cup \{e\}} Sim(x, y) = \max\left(\max_{y \in S} Sim(x, y), Sim(x, e)\right)$. To simplify the notation, let $M_x(S) = \max_{y \in S} Sim(x, y)$ represent the maximum similarity between x and the current set S. The marginal gain for a single element x becomes:

$$g_x(S) = \max(M_x(S), Sim(x, e)) - M_x(S) = \max(0, Sim(x, e) - M_x(S)). \tag{16}$$

Now, we compare $g_x(A)$ and $g_x(B)$ for $A \subseteq B$. Because A is a subset of B, we have:

$$M_x(A) \leq M_x(B). \tag{17}$$

Consider the function $h(z) = \max(0, c - z)$, where $c = Sim(x, e)$ is a constant for a fixed x and e. This function $h(z)$ is monotonically non-increasing with respect to z. Since $M_x(A) \leq M_x(B)$, it logically follows that:

$$h(M_x(A)) \geq h(M_x(B)). \tag{18}$$

Which translates to:

$$\max(0, Sim(x, e) - M_x(A)) \geq \max(0, Sim(x, e) - M_x(B)). \tag{19}$$

Since $R(x) \geq 0$, multiplying by $R(x)$ gives:

$$R(x) \cdot g_x(A) \geq R(x) \cdot g_x(B). \tag{20}$$

Summing this over all $x \in \mathcal{D}$ yields the final inequality:

$$\sum_{x \in \mathcal{D}} R(x) \cdot g_x(A) \geq \sum_{x \in \mathcal{D}} R(x) \cdot g_x(B). \tag{21}$$

Therefore, $\Delta F(e \mid A) \geq \Delta F(e \mid B)$. This proves that $F(D)$ satisfies the property of diminishing marginal returns, establishing its submodularity. ■

In light of these properties, the greedy algorithm is guaranteed to provide a $(1 - 1/e)$ approximation to the optimal subset, ensuring the theoretical robustness of our pruning strategy. This formal justification confirms that SPARK effectively balances information coverage and reasoning utility, even when navigating high-dimensional search spaces.

A.3 Detailed Algorithmic Workflow

To gain a more comprehensive and intuitive understanding of the SPARK framework, this section details its complete path-aware reasoning exploration stage algorithmic workflow. The main text primarily introduces the core concept of MCTS with our submodular-based greedy pruning algorithm, while Algorithm 1 and Figure 6 illustrates this process in greater detail. They illustrate the entire lifecycle of the autonomous reasoning exploration, explicitly highlighting how search nodes are expanded, evaluated, and selectively pruned to maximize pruning objective while strictly bounding the search width.

B Experimental Details

In this section, we will detail the specifics of the experiments presented in the main text, including the datasets §B.1, comparison methods §B.2, and implementation details of our method §B.3.

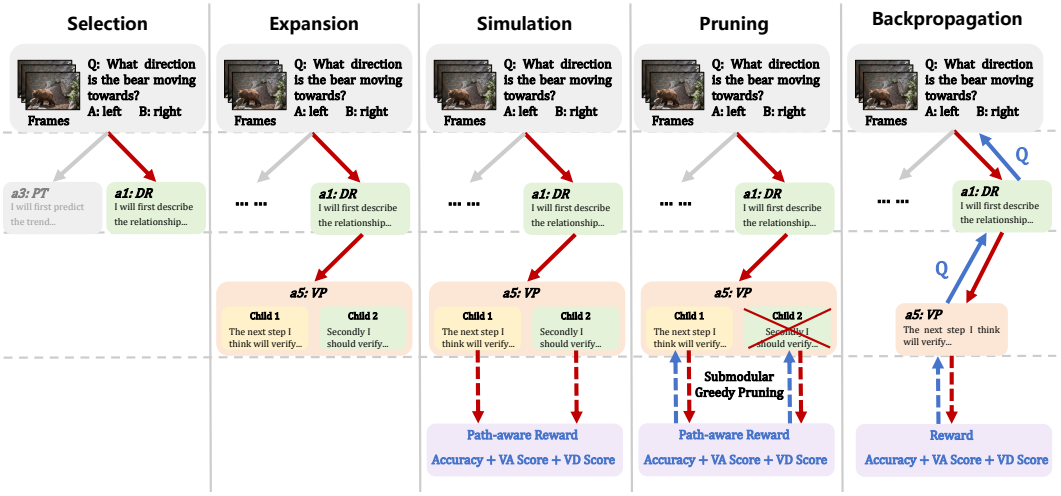


Figure 6: An illustration of the reasoning exploration stage for SPARK.

B.1 Datasets

To comprehensively evaluate the effectiveness of our method, we conduct experiments on three public spatial-temporal reasoning benchmarks, including VLM4D [11], DSI [37], and VSI [38]. These datasets cover mainstream spatialtemporal reasoning tasks including direction and rotation judgment. Table 4 summarizes the statistics of the datasets used in our main experiments, including the sample sizes of training and test splits. Notably, none of these benchmarks provide official training partitions. Therefore, we randomly sample 200 instances from each dataset with equal proportion to construct a seed training set for empirical learning of our method. Subsequently, the sampled instances are removed from the test set.

Datasets	Release	Citation	|Train|	|Test|	Tasks Type
VLM4D	2025-07	[11]	200	1616	Translation, Rotation, Counting
DSI	2025-10	[37]	200	1570	Translation, Rotation, Distance
VSI	2025-11	[38]	200	4930	Direction, Distance, Counting, Size

Table 4: Datasets in the main experiments.

B.2 Baselines

We conducted extensive experiments comparing 18 competitors, including representative models from three categories: Proprietary Models, Open-Source Models, and 3D Spatial Reasoning Models.

Proprietary Models: This category represents the current performance of general-purpose VLMs. We include the GPT series (GPT-4o [65] and GPT-5 [66]) from OpenAI and the Gemini series (Gemini-2.5-Pro [67] and Gemini-3-Pro [40]) from Google. These models are regarded as strong baselines, to evaluate whether our small-scale models with empirical self-evolution capability can match the performance of state-of-the-art industrial models.

Open-Source Models: We evaluate a wide range of popular open-source VLMs. Qwen Series: Including Qwen2.5-VL [49], Qwen3-VL [50] and Qwen3.5 [39], which are known for their strong base reasoning capabilities. InternVL Series: Specifically InternVideo2.5 [68] and InternVL3.5 [51], representing models with high-resolution vision encoders and large-scale pre-training. LLaVA Series: Including LLaVA-NeXT [52] and LLaVA-One-Vision [69], a representative architecture for efficient visual-instruction tuning.

3D Spatial Reasoning Models: These models are specifically architected or fine-tuned for spatial and temporal understanding, often through domain-specific expert distillation or reinforcement learning. VLM-3R [13] and VG-LLM [14]: Models that focus on 3D scene understanding and visual grounding. Spatial-MLLM [17] underwent instruction fine-tuning and reinforcement learning optimization specifically for understanding spatial relationships.

Prompt Template for Comparison Models

```
[{"role": system,
"content": You are an **Expert Spatial-Temporal Reasoning Model**. Your
task is to generate a **complete, step-by-step Chain-of-Thought (CoT)** to
answer the user's question, strictly utilizing the provided video context."

"-- OUTPUT FORMAT --"
"Your output **MUST** be the complete reasoning trace followed by the final
answer."
"1. The reasoning must start with "Let's think step by step."
"2. The final answer **MUST** be strictly provided in the XML tag:
<answer>The final option letter (A/B/C/D)</answer>"}

{"role": user
"image": {Video Frames}
"text": {Question}}]
```

Table 5: Prompt for the Comparison Models.

Prompt Template for Experience Extraction

```
{"role": "system",
"content": "You are a Spatio-Temporal Reasoning Strategist. Distill a
**Universal Abstract Principle** by translating actions into strategic
definitions and generalizing details into abstract concepts."}

{"role": "user",
"content": "-- STRATEGY DEFINITIONS --
- [Action List with Strategic Descriptions...]

-- CRITICAL RULES --
1. **NO ACTION NAMES**: Use bolded strategic descriptions only.
2. **STRICT GENERALIZATION**: Replace specific objects (e.g., car) with
abstract terms (e.g., dynamic object).

-- TASK --
Analyze: [Parent Context] | [Chosen Strategy] | [Rejected Strategy]
Output a single, dense heuristic sentence: 'To [Goal] in [Context], proceed
by [Chosen Action] to [Outcome], avoiding [Rejected Action] which risks
[Error].'"}
```

Table 6: Prompt for Experience Extraction.

B.3 Implementation Details

We provide experimental details in the main experiment here. For all main experimental results, we uniformly sampled the videos, limiting the maximum number of frames to 30. The system prompt used for the baseline model are shown in Table 5. During the inference phase, the max length of tokens is set to 8192 and the Temperature is set to 0.9. In reasoning exploration stage, we configure the MCTS to run for a total of 30 iterations per sample. For each expansion step, the VLM generates $|\mathcal{C}| = n = 8$ candidate sibling nodes. To evaluate these candidates, we perform 4 simulations for each node. Following the pruning process, we retain the top 4 nodes ($|D| = m = 4$) for subsequent tree growth. To construct the contrastive dataset $\mathcal{P}$, we set the threshold λ to 0.75 and α to 0.30. In path-aware experience extraction stage, we set the temperature to 0.7 and top_p to 0.9 to encourage the generation of diverse insights. The specific prompts for experience extraction and improvement are presented in Table 6 and Table 7. During the path-aware experience utilization stage, we employ the all-MiniLM-L6-v2 [72] sentence transformer to encode the video-query contexts for semantic similarity search. Regarding the experimental setup, all experimental results are the average results of 5 different random seeds, and the experiments are conducted on three NVIDIA A100 GPUs.

Prompt Template for Experience Refinement

```
{"role": "system",
"content": "You are a Spatio-Temporal Knowledge Base Administrator.
Integrate a NEW INSIGHT into the KB while maintaining conciseness and
distinctness."}

{"role": "user",
"content": "-- CURRENT KB --
[Index]: [Principle]...

-- NEW INSIGHT --
{new_insight}

-- DECISION RULES --
1. **merge**: Rewrite existing principle if insight is similar.
2. **add**: Propose as a new principle if novel.
3. **ignore**: Return empty array if redundant.

-- STRICT JSON OUTPUT FORMAT --
- ADD: [{"option": "add", "experience": "..."}]
- MERGE: [{"option": "merge", "experience": "...", "merged_from": ["idx"]}]
- IGNORE: []"}
```

Table 7: Prompt for Experience Refinement.

C Additional Results

In this section, we supplement additional experimental results and method analyses, covering six core dimensions:cross-domain generalization §C.1, inference time scaling §C.2, comparison with other search methods §C.3, computational cost §C.4, saliency analysis, and case studies §C.6, so as to further verify the effectiveness and superiority of our method.

C.1 Analysis on Cross-Domain Generalization

In this section, we explore whether the experiences obtained by SPARK can achieve cross-dataset generalization. Specifically, we take VLM4D [11], DSI [37], and VSI [38] as target datasets respectively, and use the experiences from the other two datasets to guide the reasoning of VLMs. The experimental results are shown in Table 8. The experimental results demonstrate that the experiences obtained by SPARK can achieve cross-domain generalization. Under all experimental settings, the model achieves better performance than the Vanilla model after being guided by experiences. This result reveals that the experiences obtained by the SPARK method have universal ability to guide the model in enhancing its spatialtemporal reasoning capability.

Model Type	Methods	D+V→M	V+M→D	D+M→V	Average
Qwen3.5-9b[39]	Vanilla	65.91	44.30	54.87	55.03
	SPARK	68.21	48.66	60.48	59.11
Qwen3-vl-8b[50]	Vanilla	58.98	41.47	42.15	47.53
	SPARK	60.72	42.11	49.34	50.72
InternVL3.5-8B[51]	Vanilla	42.23	36.42	35.05	37.90
	SPARK	44.09	40.30	38.21	40.87
LLaVA-NeXT-7B[52]	Vanilla	36.22	33.32	30.08	33.21
	SPARK	41.69	36.22	33.37	37.09

Table 8: Analysis on Cross-Domain Generalization [4].

C.2 Analysis on Inference Time Scaling

In this section, we explore the impact of inference time scaling on the performance of our method. Specifically, during the inference process, we gradually increase the number of sample-specific sub-bank $\mathcal{E}_{(V^*,Q^*,Y^*)}$ obtained in the adaptive experience retrieval stage. As shown in Figure 7, under all settings, introducing more experiences as reasoning guidelines leads to a continuous improvement in method performance.

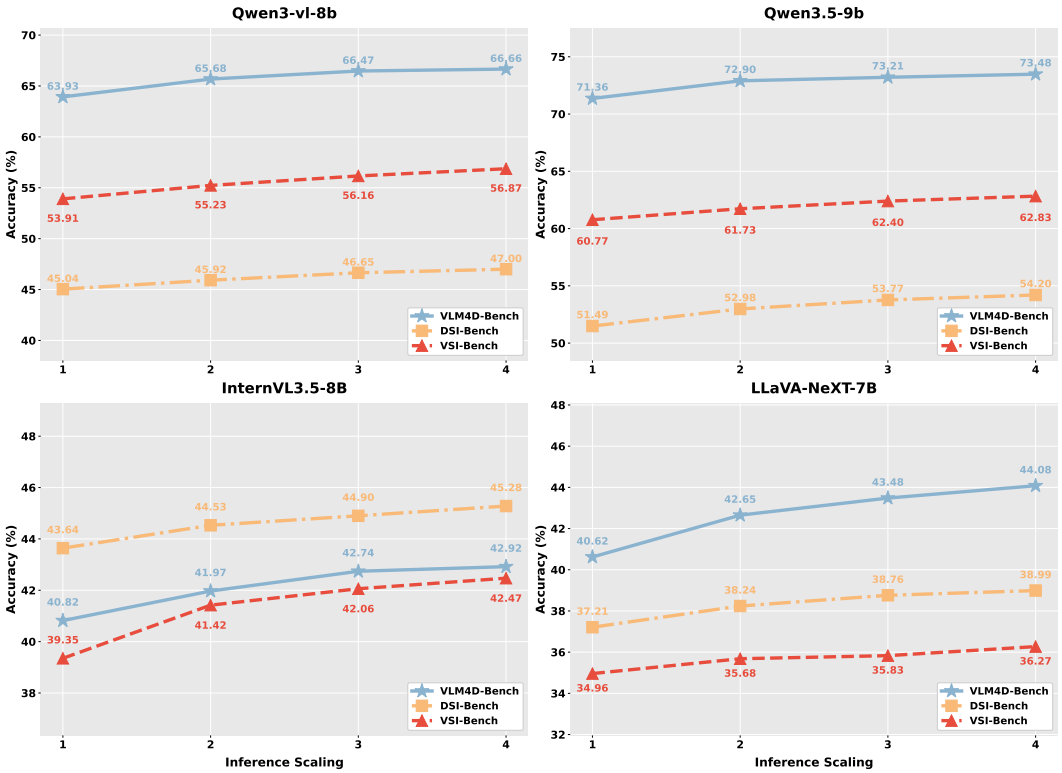


Figure 7: Analysis on Inference Time Scaling.

C.3 Comparison with Other Search Based Methods

In this section, we present a comparative analy- sis of UCoT and existing search based or experiential multimodal reasoning meth- ods. Specifically, the baseline methods we compared include: Mulberry[70], LlamaV-o1[62], LLaVA-CoT[59], AR-MCTS[71] and Astar[8]. The results are presented in Table 9. Experimental results demonstrate that, in most settings, our SPARK method is able to more effectively integrate the inherent reasoning capabilities of VLMs with external experience, thereby achieving exceptional spatiotemporal reasoning performance. For example, on the challenging VLM4D dataset, SPARK outperforms Astar, the outcome-based experiential method that achieves the second-best performance, by 2.48%. This result highlights the contribution of using visual information as a reward: only when failure details are finely perceived can they provide useful guidance for future decision-making.

Model	VLM4D	DSI	VSI	Average
Vanilla[39]	65.91	44.30	54.87	55.03
Mulberry[70]	67.96	45.14	55.67	56.26
LlamaV-o1[62]	66.87	44.71	55.52	55.70
LLaVA-CoT[59]	65.11	45.70	56.86	55.89
AR-MCTS[71]	67.64	50.40	59.41	59.15
Astar[8]	68.88	48.72	58.85	58.82
SPARK	**71.36**	**51.49**	**60.77**	**61.17**

Table 9: Performance comparison with other search based methods using Qwen3.5-9b as backbone.

[4]For the sake of simplicity, we denote datasets VLM4D [11], DSI [37], and VSI [38] by the letters M, D, and V, respectively.

C.4 Analysis on Computational Cost

Standard MCTS-style reasoning search often brings substantial computational overhead, while training-based spatio-temporal multimodal methods demand heavy time investment for parameter tuning. In this section, we analyze the computation-accuracy trade-off of SPARK by quantifying its preparation cost under identical hardware settings. Specifically, we adopt preparation time as the core efficiency metric: for training-based baselines, this corresponds to full-model training time, whereas for search-based methods, it denotes the time cost for trajectory exploration over the benchmark samples. We pair this metric with the average prediction accuracy across three spatio-temporal benchmarks to jointly compare overall performance. As shown in Table 10, SPARK achieves a favorable balance between computation cost and final accuracy, reaching an average accuracy of 61.17% with only 3.2 hours of preparation time. It matches the fast search baseline Astar in wall-clock time (3.2 h) while obtaining a clear accuracy gain, and surpasses AR-MCTS by 2.02% in accuracy despite drastically shorter preparation time. Compared with training-based approaches including VLM-3R and Spatial-MLLM, SPARK cuts the preparation time from 27.3–34.3 hours down to 3.2 hours and delivers much stronger prediction performance. These observations demonstrate that our experience-library-based MCTS design introduces only a one-time upfront overhead; after library construction, inference remains lightweight, training-free, and well suited for scalable spatio-temporal reasoning.

Model	Prep. Time (h)	Avg Acc. (%)
VLM-3R[13]	34.3	44.14
Spatial-MLLM[17]	27.3	42.60
AR-MCTS[71]	16.4	59.15
Astar[8]	3.2	58.82
SPARK	**3.2**	**61.17**

Table 10: Efficiency comparison with baseline methods using Qwen3.5-9B as backbone.

C.5 Analysis on Statistical Saliency

To rigorously verify that the performance gains observed in our experiments are statistically significant and not due to random variation, we perform the Wilcoxon Signed-Rank Test [73]. As a non-parametric statistical hypothesis test, it is particularly suitable for comparing paired performance metrics (i.e., VLM performance with vs. without SPARK) across various benchmarks without assuming a normal distribution of the data.

Hypothesis Setting. We test the null hypothesis (H_0) that there is no significant difference in reasoning accuracy between the baseline VLMs and those empowered by the SPARK framework. The alternative hypothesis (H_1) posits that the incorporation of our process-aware experience paradigm leads to a statistically significant improvement in spatiotemporal reasoning.

Results Analysis and Conclusion. We adopt a standard significance level of $\alpha = 0.05$. As summarized in Table 11, we report the calculated p-values for four backbone models (Qwen3-VL-8B, Qwen3.5-9B, InternVL3.5-8B, and LLaVA-NeXT-7B) across three primary benchmarks. The empirical results show that all calculated p-values are significantly below the 0.05 threshold. Notably, on the VLM4D-Bench, p-values for all models remain in the order of 10^{-3} to 10^{-4}, indicating a highly robust performance leap. Since all p-values across all tested dimensions are consistently less than α, we reject the null hypothesis. These findings provide strong statistical evidence that SPARK consistently and significantly enhances the spatiotemporal reasoning capabilities of diverse VLMs.

Model	**VLM4D-Bench**	**DSI-Bench**	**VSI-Bench**
Qwen3-VL-8B [50]	2.86×10^{-4}	2.17×10^{-3}	5.40×10^{-3}
Qwen3.5-9B [39]	4.88×10^{-4}	5.01×10^{-3}	8.77×10^{-3}
InternVL3.5-8B [51]	4.63×10^{-4}	3.11×10^{-2}	3.62×10^{-4}
LLaVA-NeXT-7B [52]	1.58×10^{-3}	7.29×10^{-3}	3.45×10^{-4}

Table 11: Statistical significance results ($\alpha = 0.05$).

C.6 Case Study

We conduct a case study with Qwen3.5-9B on the VLM4D, DSI, and VSI datasets to inspect model reasoning behavior. We visualize region attribution C_i^j and token-wise VD metric for reasoning paths of varying quality, as shown in Figure 8–Figure 13.

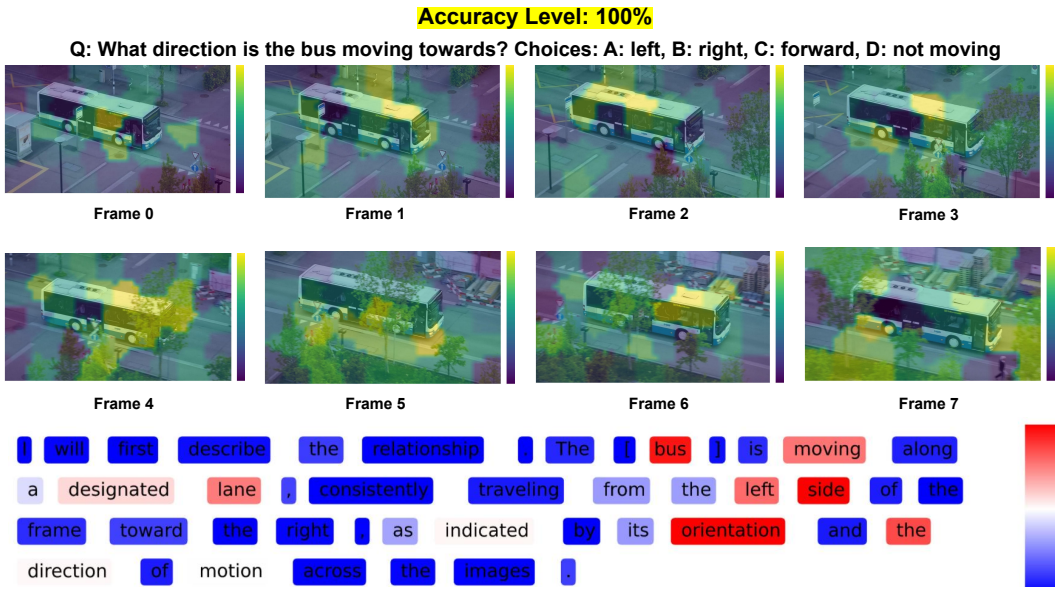


Figure 8: A case study on VLM4D dataset using Qwen3.5-9B.

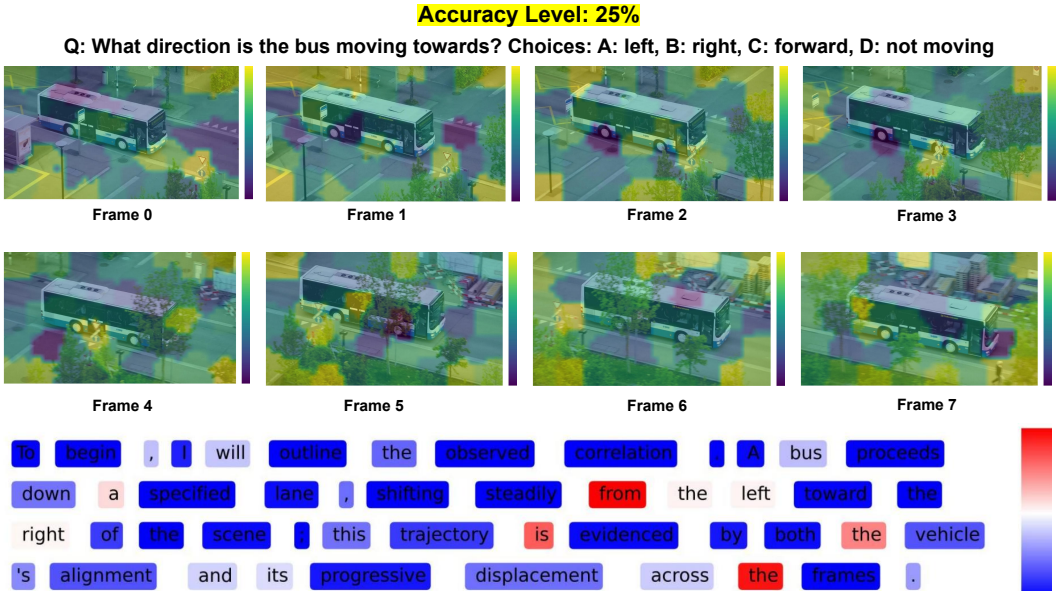


Figure 9: A case study on VLM4D dataset using Qwen3.5-9B.

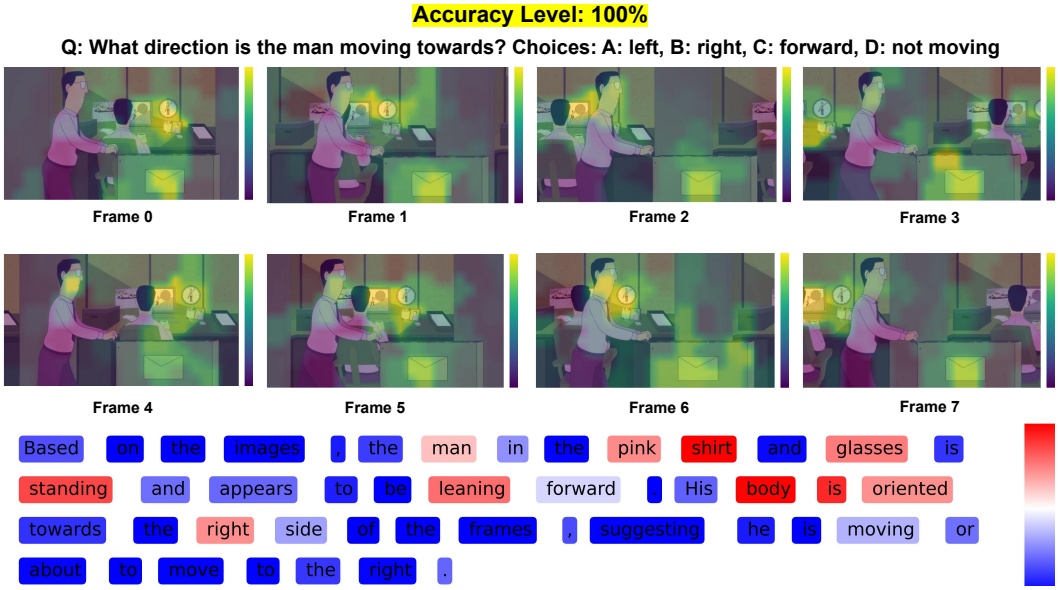


Figure 10: A case study on DSI dataset using Qwen3.5-9B.

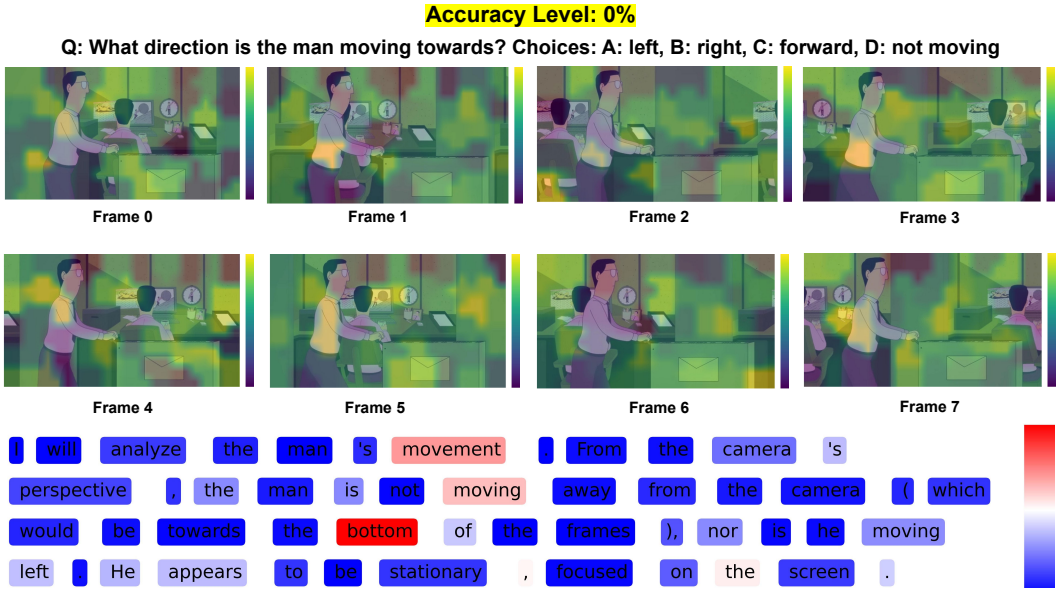


Figure 11: A case study on DSI dataset using Qwen3.5-9B.

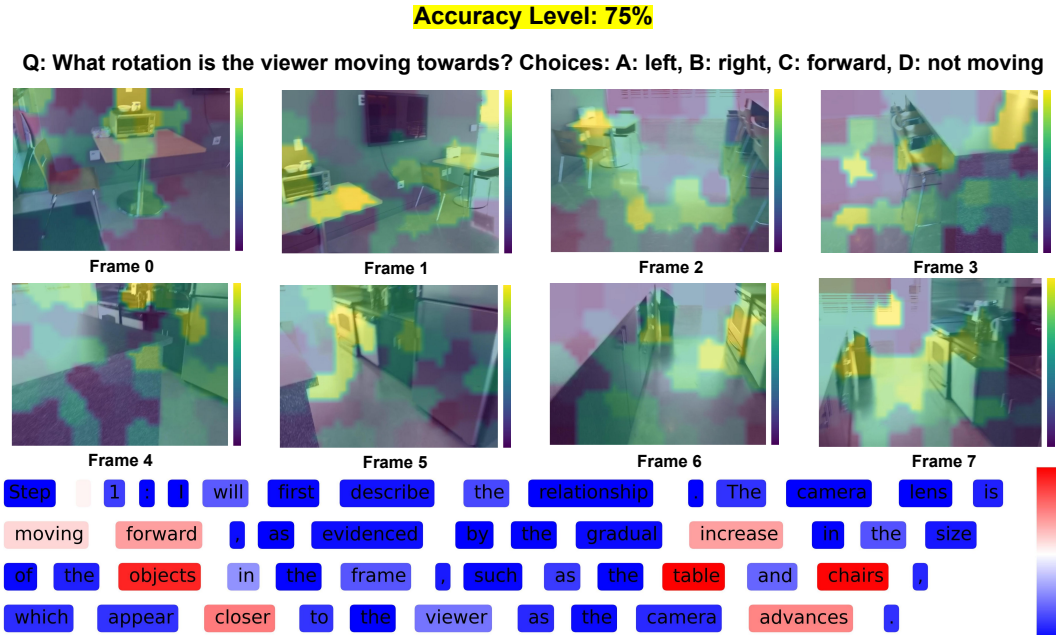


Figure 12: A case study on VSI dataset using Qwen3.5-9B.

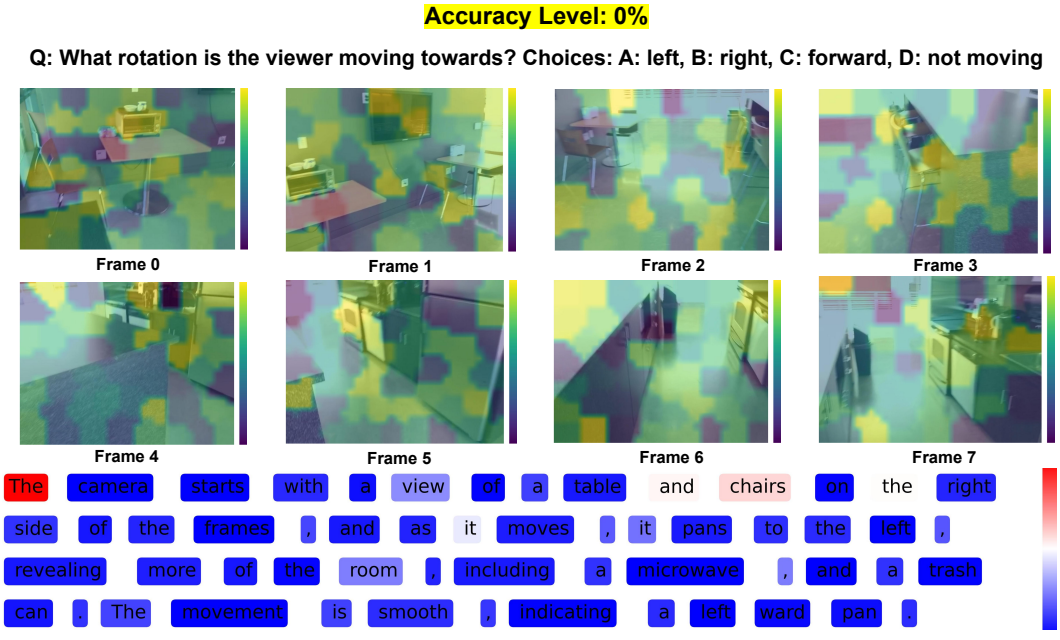


Figure 13: A case study on VSI dataset using Qwen3.5-9B.

Algorithm 1 Path-aware Reasoning Exploration

Input: Seed dataset $\mathbb{D}$, VLM VLM$(\cdot)$, Action set $\mathcal{A} = \{a_1, \ldots, a_7\}$, Exploration budget T, Expansion size n, Pruning size m, Contrastive threshold α.
Output: High-quality reasoning path set $\mathcal{P}$.
Initialize action space$A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$
Initialize repository $\mathcal{P} \leftarrow \emptyset$
for each sample $(V, Q, Y) \in \mathbb{D}$ **do**
    Initialize tree $\mathcal{T}$ with root $S_0 = (V, Q)$;
    for $iter = 1$ **to** T **do**
        // (1) Selection
        $S_{curr} \leftarrow S_0$;
        while S_{curr} is not a leaf node **do**
            $r^* \leftarrow \arg\max_{r \in \text{children}(S_{curr})} UCT(r)$;
            $S_{curr} \leftarrow S_{curr} \cup \{r^*\}$;
        end while
        // (2) Expansion
        $\mathcal{C} \leftarrow \emptyset$;
        $a \leftarrow$ Sample from $\mathcal{A}$;
        for $i = 1$ **to** n **do**
            $r_{t,i} \sim \text{VLM}(S_{curr}, a)$ ▷ Generate candidate step;
            $\mathcal{C} \leftarrow \mathcal{C} \cup \{r_{t,i}\}$;
        end for
        // (3) Simulation
        for each candidate $r_{t,j} \in \mathcal{C}$ **do**
            Execute simulation rollouts to obtain path-aware reward $R(r_{t,j})$ ▷ §4.1;
        end for
        // (4) Greedy Pruning (Submodular Maximization)
        $D \leftarrow \emptyset$;
        while $|D| < m$ **do**
            $r^* \leftarrow \arg\max_{r \in \mathcal{C} \setminus D} (F(D \cup \{r\}) - F(D))$ ▷ Marginal gain;
            $D \leftarrow D \cup \{r^*\}$;
        end while
        Append D to $\mathcal{T}$ as children of S_{curr};
        Prune $\mathcal{C} \setminus D$;
        // (5) Backpropagation
        Update $Q(r)$ and $N(r)$ for all nodes in the path from D to S_0;
    end for
    // (6) Contrastive Pair Harvesting
    for each internal node $S \in \mathcal{T}$ **do**
        $\mathcal{K} \leftarrow$ children(S);
        for each pair $(r_i, r_j) \in \mathcal{K} \times \mathcal{K}$ **do**
            if $Q(r_i) - Q(r_j) > \lambda$ **then**
                $\mathcal{P} \leftarrow \mathcal{P} \cup \{(r_i, r_j)\}$ ▷ Collect sibling pairs;
            end if
        end for
    end for
end for
Return $\mathcal{P}$

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 main claims regarding the SPARK framework, its theoretical bounds, and empirical performance are stated in the Abstract and §1, and are fully supported by theoretical proofs in Appendix A.2 and experiments in §5.

   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 explicitly discuss the limitations of our work in §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: Our work is application based and doesn't include theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We provide all the information to replicate the main experiment of the paper in Appendix B

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We provide all required code and datasets for this work in the supplementary material.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: All training and testing details necessary to understand the results are provided in Appendix B.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification: We show the random seed settings of the main experiment of this article in Appendix B.3 (Line number: 690) Furthermore, we also conducted a statistical saliency analysis of the results in Appendix C.5.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We introduce the types of computer workers in Appendix B.3 (Line Number: 690).

   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 checked that our work complies with the NeurIPS Code of Ethics.

   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: [N/A]

    Justification: Our work is conducted on general spatiotemporal tasks in the CV field and has no social impact.

    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: This paper poses no such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We cite all datasets (§5 Line Number: 236 and Appendix B.1) and models (Appendix B.2) used in this article.

    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 release new assets.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- 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: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    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.