PANDA: Towards Generalist Video Anomaly Detection via Agentic AI Engineer

Zhiwei Yang[1,2] **Chen Gao**[2†] **Mike Zheng Shou**[2†]
[1]Xidian University [2]Show Lab, National University of Singapore

Abstract

Video anomaly detection (VAD) is a critical yet challenging task due to the complex and diverse nature of real-world scenarios. Previous methods typically rely on domain-specific training data and manual adjustments when applying to new scenarios and unseen anomaly types, suffering from high labor costs and limited generalization. Therefore, we aim to achieve generalist VAD, *i.e.*, automatically handle any scene and any anomaly types without training data or human involvement. In this work, we propose PANDA, an agentic AI engineer based on MLLMs. Specifically, we achieve PANDA by comprehensively devising four key capabilities: (1) self-adaptive scene-aware strategy planning, (2) goal-driven heuristic reasoning, (3) tool-augmented self-reflection, and (4) self-improving chain-of-memory. Concretely, we develop a self-adaptive scene-aware RAG mechanism, enabling PANDA to retrieve anomaly-specific knowledge for anomaly detection strategy planning. Next, we introduce a latent anomaly-guided heuristic prompt strategy to enhance reasoning precision. Furthermore, PANDA employs a progressive reflection mechanism alongside a suite of context-aware tools to iteratively refine decision-making in complex scenarios. Finally, a chain-of-memory mechanism enables PANDA to leverage historical experiences for continual performance improvement. Extensive experiments demonstrate that PANDA achieves state-of-the-art performance in multi-scenario, open-set, and complex scenario settings without training and manual involvement, validating its generalizable and robust anomaly detection capability. Code is released at https://github.com/showlab/PANDA.

1 Introduction

Video anomaly detection (VAD) [1, 2, 3, 4] aims to identify abnormal or suspicious events in video streams, playing a vital role in a wide range of real-world applications such as intelligent surveillance [5], traffic monitoring [6], autonomous driving [7], and industrial safety [2].

Existing VAD methods follow a specialist-oriented paradigm and require manual participation when deploying for new scenarios and anomalies. Broadly, they can be categorized into: training-dependent and training-free (Fig. 1(a)). Specifically, training-dependent methods rely on newly annotated data to train models for each target scenario. The manual and training costs make such methods lack generalization and versatility. Besides, training-free methods typically employ pre-trained large language models (LLMs) or vision-language models (VLMs) as the backbone, thereby eliminating the need for model training. However, they still depend heavily on manual engineering when deploying for new scenarios and anomalies, such as scenario-specific preprocessing steps, handcrafted prompt templates, rule curation, and post-processing. These static pipelines still lack adaptivity, making them brittle when confronted with uncertainty, long-term temporal dependencies, or complex, dynamic scenarios. Moreover, the hand-crafted nature restricts them from towarding generalist VAD.

†Corresponding authors.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

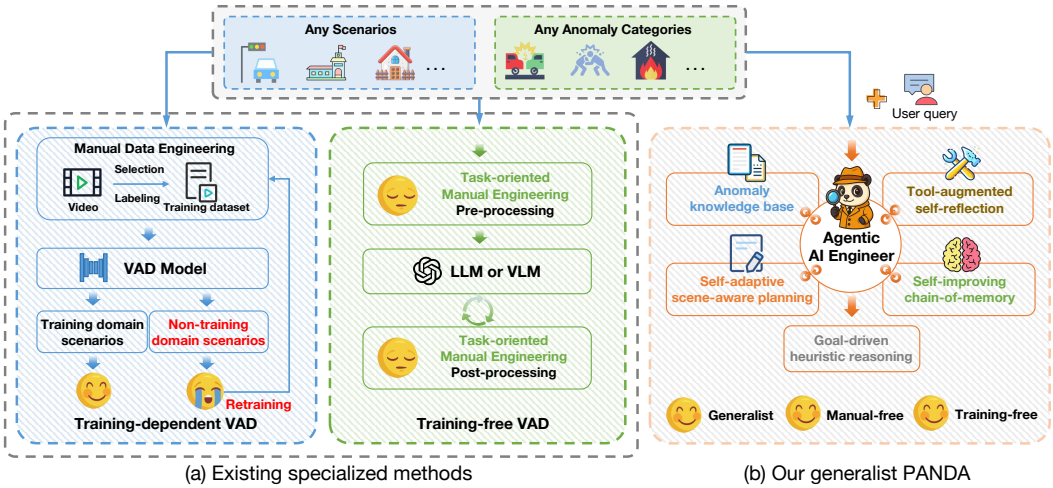

Figure 1: **PANDA vs. Existing specialized methods.** When facing arbitrary scenes and anomalies, PANDA can automatically adapt without the need for training refinements and manual adjustments, still achieving superior performance.

To overcome the limitations of existing methods and free ourselves from the burden of domain-specific training and handcrafted pipeline design, our vision is to develop a general-purpose video anomaly detection method that can be self-adaptive to new scenarios and novel anomaly types without requiring any training data or manual pipeline assembly. The recent success of Multimodal Large Language Models (MLLMs) in a wide range of visual understanding tasks offers a promising foundation for realizing this vision. Therefore, we introduce PANDA, an agentic AI engineer for generalized VAD (Fig. 1(b)). Drawing inspiration from how human engineers systematically analyze problems, adapt to complex environments, and iteratively improve through tool use and experience, PANDA adaptively perceives the enviroment based on user-defined requirements, formulates detecion plans, perform goal-driven reasoning, invokes external tools to enhance decsion making, and continuously accumulates expeirence in memory for self-improvement.

Technically, the proposed PANDA is distinguished by the following aspects. ***(1) self-adaptive scene-aware strategy planning.*** Faced with a new scene or user-defined anomaly detection requirements, PANDA first conducts environment perception and understanding, then retrieves relevant anomaly rules from an anomaly knowledge database. Based on the environment context information, a scene-adaptive Retrieval Augmented Generation (RAG) mechanism is designed to construct tailored anomaly detection plans. ***(2) Goal-driven heuristic reasoning.*** PANDA injects task-specific prompts guided by latent anomaly cues, which steer the reasoning process toward more accurate and focused decision-making. ***(3) Tool-augmented self-reflection.*** PANDA iteratively assesses uncertainty and activates a suite of curated tools, such as object detection, image retrieve, or web search, to acquire additional information and resolve ambiguous decision-making. ***(4) Self-improving chain-of-memory.*** PANDA integrates historical experiences to justify current reasoning decisions or self-reflection. By progressively accumulating contextual cues across temporal spans, it enhances both the stability and accuracy of its decisions over time. Taken together, PANDA embodies an agentic AI engineer that proactively perceives diverse environments, formulates adaptive strategies, performs goal-driven reasoning, and progressively improves through tool-augmented reflection and a chain-of-memory mechanism, enabling generalizable video anomaly detection across complex real-world scenarios.

Extensive experiments across multiple challenging benchmarks show that PANDA achieves state-of-the-art performance in multi-scenario, open-set, and complex scenario settings, *without training and manual involvement*. These results highlight PANDA's strong potential as an autonomous and general-purpose solution for real-world VAD.

2 Related Work

Video anomaly detection (VAD) [8, 9, 10, 11], has long been a critical research topic in the computer vision field due to its wide range of real-world applications. Existing VAD methods are specialist-oriented and can be broadly categorized into training-dependent and training-free approaches.

Training-dependent VAD. These approaches rely on varying levels of annotated data and typically fall into three categories: semi-supervised VAD [2, 5, 10, 12, 13], weakly-supervised VAD [14, 15, 16, 17, 18, 19], and instruction-tuned VAD [20, 21, 22, 23, 24]. For example, Ristea et al. [25] proposed an efficient anomaly detection model based on a lightweight masked autoencoder. Yang et al. [14] introduced a text prompt-driven pseudo-labeling and self-training framework for weakly-supervised VAD. Zhang et al. [23] presented a model combining an anomaly-focused temporal sampler with an instruction-tuned MLLM to detect anomalies. While these training-dependent methods often perform well within the domain of the training data, they typically suffer from sharp performance degradation when deployed in out-of-distribution environments or faced with novel anomaly types. This limits their applicability in the open-world scenarios where anomalies are diverse, and context-sensitive.

Training-free VAD. Inspired by the recent success of LLMs [26, 27] and VLMs [28, 29, 30], training-free VAD methods [31, 32] have gained increasing attention. These approaches aim to leverage the powerful prior knowledge embedded in foundation models without requiring domain-specific training. For instance, Zanella et al. [31] proposed the first language-model-based training-free VAD framework, which improves anomaly scoring by aligning cross-modal features between LLMs and VLMs while suppressing noisy captions. Yang et al. [32] developed a rule-based anomaly inference framework by prompting LLMs to perform inductive and deductive reasoning over anomaly rules. Despite removing the need for training, these methods often rely on static prompting patterns and require substantial manual engineering (*e.g.*, handcrafted pre/post-processing), which limits their adaptivity and robustness in complex, real-world scenarios.

Distinct from both paradigms above, PANDA is an agent-based framework that embodies the characteristics of an agentic AI engineer, which is capable of autonomously performing VAD without training and manual engineering when faced with various real-world scenarios. By incorporating a progressive reflection mechanism and a suite of perception-enhancing tools, PANDA can adaptively refine its predictions through self-reflection and tool invocation. This enables PANDA to dynamically handle diverse and challenging scenarios in the real world.

3 Method

In this section, we present the core architecture and reasoning process of PANDA, an agentic AI engineer for generalized VAD. PANDA is designed to dynamically perceive diverse environments and perform progressive, tool-enhanced reasoning and self-refinement, as shown in Fig. 2. It achieves this through four synergistic modules: (1) self-adaptive scene-aware strategy planning, (2) goal-driven heuristic reasoning, (3) tool-augmented self-reflection, and (4) self-improving chain-of-memory.

3.1 Self-adaptive Scene-aware Strategy Planning

To achieve VAD in general and unconstrained environments, it is essential to dynamically perceive the current video context and construct targeted detection strategies. Given the scene-dependence of many real-world anomalies and the variability in visual conditions, PANDA first performs self-adaptive perception of the input video to extract high-level environment contextual information.

Environmental Perception. Given a user-defined detection query $\texttt{User}_{\text{query}}$ and an input video sequence $V = \{f_1, f_2, \ldots, f_N\}$ containing N frames, PANDA uniformly samples M keyframes $F = \{f_1, f_2, \ldots, f_M\}$ and constructs a perception prompt $\texttt{Prompt}_{\text{perception}}$ combining F with the $\texttt{User}_{\text{query}}$. This prompt is fed to a VLM, which returns structured environmental information:

$$\begin{aligned} \texttt{EnvInfo} &= \text{VLM}(F, \texttt{Prompt}_{\text{perception}}) \\ &= \{\texttt{Scene Overview}, \texttt{Potential Anomalies}, \\ &\quad \texttt{Weather Condition}, \texttt{Video Quality}\} \end{aligned} \tag{1}$$

Here, `Scene Overview` provides a high-level summary of the scene, including location type (*e.g.*, street, shop, parking lot) and observed activities. `Potential Anomalies` refers to types of suspicious behaviors that may plausibly occur in the current scene context. `Weather Condition` captures attributes such as time of day (day/night) and weather (*e.g.*, sunny, rainy). `Video Quality` summarizes resolution and clarity (*e.g.*, low-resolution, blurred, noisy).

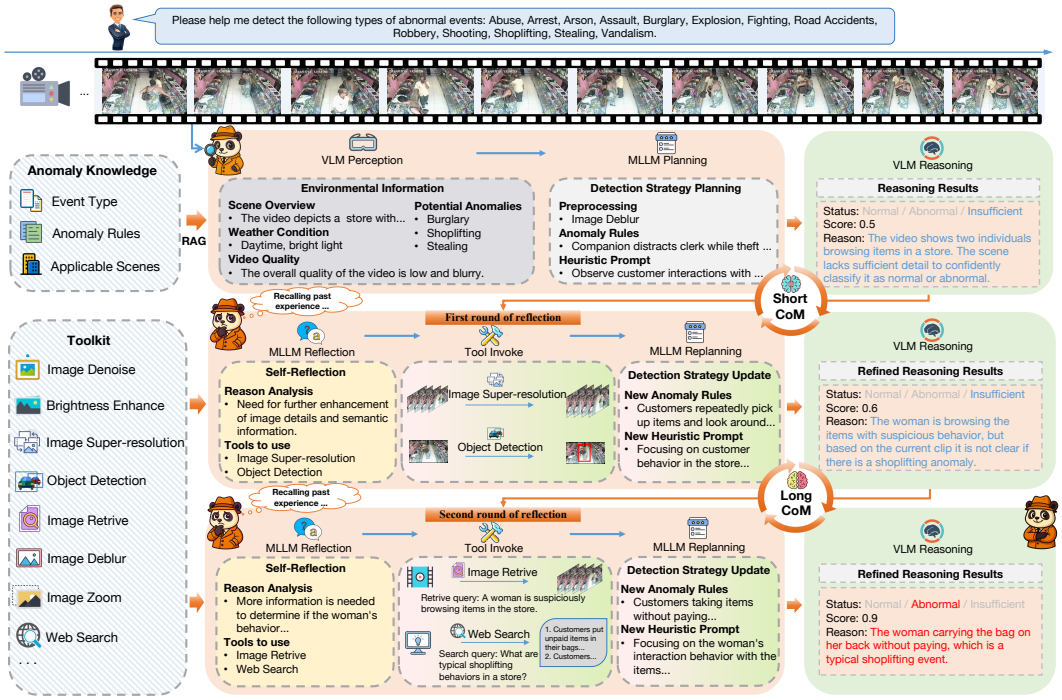

Figure 2: **Overview of the proposed PANDA.** As shown in the figure, upon receiving a user-defined query, PANDA first performs environment perception and plans a scene-adaptive detection strategy. PANDA then executes the plan with goal-driven heuristic reasoning. When encountering ambiguous cases, PANDA enters a reflection phase, revising its plan and invoking external tools to refine the decision. Throughout the process, PANDA maintains both short-term and long-term Chain-of-Memory (CoM), enabling it to accumulate experience and continually improve over time.

RAG-Based Strategy Planning. With the structured environment context in hand, PANDA proceeds to plan its detection strategy. To avoid hallucinations and improve reliability, this planning is performed via retrieval-augmented generation (RAG) [33], driven by a multimodal large language model (MLLM). First, based on $\texttt{User}_{\text{query}}$, PANDA constructs a knowledge base prompt $\texttt{Prompt}_{\text{know}}$ and then generates a structured general anomaly knowledge base using the MLLM:

$$\kappa_a = \text{MLLM}(\texttt{User}_{\text{query}}, \texttt{Prompt}_{\text{know}}) = \{\texttt{Event Type}, \texttt{Anomaly Rules}, \texttt{Application Scenes}\}. \tag{2}$$

Here, `Event Type` indicates anomaly categories specified by the user. `Anomaly Rules` are detection rules associated with each anomaly type. `Application Scenes` are contextual environments where anomalies are likely to occur.

For each anomaly type, we predefine H rule-scene pairs to form the knowledge base. PANDA then uses the perceived `EnvInfo` as a query to retrieve the top-k most relevant anomaly rules:

$$\texttt{Rules}_a = \texttt{RetrieveTopK}(\texttt{EnvInfo}, \kappa_a). \tag{3}$$

Finally, PANDA integrates the $\texttt{User}_{\text{query}}$, `EnvInfo`, and $\texttt{Rules}_a$ to construct a planning prompt $\texttt{Prompt}_{\text{plan}}$, which is passed to the MLLM to generate the detection strategy plan:

$$\begin{aligned} \texttt{Plan}_{\text{strategy}} &= \text{MLLM}(\texttt{Prompt}_{\text{plan}}) \\ &= \{\texttt{Preprocessing}, \texttt{Potential Anomalies}, \\ &\quad \texttt{Heuristic Prompt}\}. \end{aligned} \tag{4}$$

Here, `Preprocessing` specifies optional visual enhancement steps (*e.g.*, brightness adjustment, denoising, super-resolution). `Potential Anomalies` refines the anomaly list based on rule relevance and scene understanding. `Heuristic Prompt` includes step-by-step reasoning instructions for each potential anomaly, enabling the downstream inference module to perform structured, chain-of-thought analysis.

By integrating self-adaptive environment perception and RAG-enhanced strategy planning, PANDA ensures that subsequent anomaly reasoning is goal-driven and context-aware, significantly improving robustness in open-world settings.

3.2 Goal-Driven Heuristic Reasoning

The reasoning module serves as the core component of PANDA for analyzing video anomaly events. PANDA supports both offline and online inference modes. In this section, we focus on the offline setting, while the implementation details section will describes the online mode settings.

Under the guidance of the detection strategy plan constructed in subsection 3.1, PANDA performs goal-driven heuristic reasoning using a VLM. Given the $\texttt{User}_{\text{query}}$, a clip-level video segment $V_{\text{clip}} = \{c_1, c_2, \ldots, c_T\}$ (each video clip c_t contains s video frames), and the $\texttt{Plan}_{\texttt{strategy}}$, PANDA first applies the preprocessing tools specified in $\texttt{Plan}_{\texttt{strategy}}$ to obtain an enhanced video clip:

$$\widetilde{V}_{\text{clip}} = \text{Preprocessing}(V_{\text{clip}}) = \{\widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_T\}. \tag{5}$$

Next, PANDA constructs a reasoning prompt based on the $\texttt{Plan}_{\texttt{strategy}}$:

$\texttt{Prompt}_{\texttt{reasoning}}$ = {$\texttt{Memory}_{\texttt{text}}^{l\text{-steps}}$, Potential Anomalies, $\texttt{Rules}_a$, Heuristic Prompt, Enhancement and Reflection Info}.

The fields Potential Anomalies, $\texttt{Rules}_a$, and Heuristic Prompt are directly inherited from the planning stage. The Enhancement and Reflection Info field incorporates information produced during the self-reflection stage (To be described in subsection 3.3), including tool-based refinements and updated anomaly rules and heuristic prompts. To enhance temporal awareness, PANDA equips a short-term memory component $\texttt{Memory}_{\text{text}}^{l\text{-steps}}$, which records the past l reasoning steps as textual memory. In addition to textual memory, PANDA also maintains a corresponding visual memory stream $\texttt{Memory}_{\text{visual}}^{l\text{-steps}}$, which stores visual frames aligned with the latest l steps, allowing the model to access fine-grained visual cues during inference.

Finally, driven by the potential anomaly targets and enriched contextual knowledge, PANDA performs heuristic reasoning with the following formulation:

$$\begin{aligned} \texttt{Result}_{\text{reasoning}} &= \text{VLM}(\widetilde{c}_t, \texttt{Memory}_{\text{visual}}^{l\text{-steps}}, \texttt{Prompt}_{\text{reasoning}}) \\ &= \{\texttt{Status} : \texttt{Normal}/\texttt{Abnormal}/\texttt{Insufficient}, \\ &\quad \texttt{Score} \in [0,1],\ \texttt{Reason} : \langle\cdot\rangle\}. \end{aligned} \tag{6}$$

Here, Status indicates the result of the VLM judgment: Normal indicates the clip is confidently classified as non-anomalous, Abnormal denotes strong evidence of anomaly, and Insufficient suggests the current information is inadequate to make a definitive judgment. Score is the probability of the existence of an abnormal event for the clip corresponding to each status. Reason is the reason for the status judgment given by the VLM. When the result is Insufficient, PANDA will trigger the reflection mechanism to gather additional context or observations before re-entering the reasoning loop.

3.3 Tool-Augmented Self-Reflection

In complex scenarios, PANDA may not be able to make a clear decision on whether a video segment is normal or abnormal. In such ambiguous cases, it returns an Insufficient status, which triggers the reflection module. PANDA adopts a tool-augmented self-reflection mechanism enhanced by a specialized set of tools $\tau = \{\texttt{tool}_1, \texttt{tool}_2, \ldots, \texttt{tool}_n\}$ for visual content enhancement and auxiliary analysis, including image deblurring, denoising, brightness enhancement, image retrieval, object detection, and web search, etc. These tools assist in gathering additional evidence to support the decision-making process.

Experience-Driven Reflection. Given an Insufficient Reason from the current reasoning step, PANDA first queries its long chain-of-memory (Long CoM, will be introduced in 3.4) to retrieve the most similar history reflection cases:

$$\texttt{Experience}_{\text{reflection}} = \texttt{RetrieveTop1}(\texttt{Insufficient Reason}, \texttt{Long CoM}). \tag{7}$$

PANDA then constructs a reflection prompt using video context information, including the $\texttt{User}_{\text{query}}$, $\texttt{EnvInfo}$, $\texttt{Plan}_{\texttt{strategy}}$, $\texttt{Rules}_a$, short chain-of-memory (short CoM), $\texttt{Insufficient Reason}$, and $\texttt{Experience}_{\text{reflection}}$: $\texttt{Prompt}_{\text{reflection}} = \{\texttt{User}_{\text{query}}, \texttt{EnvInfo}, \texttt{Plan}_{\texttt{strategy}}, \texttt{Rules}_a, \text{short CoM}, \texttt{Insufficient Reason}, \texttt{Experience}_{\text{reflection}}\}$. This prompt is fed into the MLLM to analyze the cause of uncertainty and recommend an appropriate reflection plan:

$$\begin{aligned}\texttt{Result}_{\text{reflection}} &= \text{MLLM}(\texttt{Prompt}_{\text{reflection}}) \\ &= \{\texttt{"Insufficient Reason"} : \langle\cdot\rangle, \\ &\quad \texttt{"Tools to Use"} : [\{\texttt{tool}_1, \texttt{params}\}, \ldots, \{\texttt{tool}_n, \texttt{params}\}], \\ &\quad \texttt{"New Anomaly Rule"} : \langle\cdot\rangle, \\ &\quad \texttt{"New Heuristic Prompt"} : \langle\cdot\rangle\}.\end{aligned} \tag{8}$$

Here, $\texttt{Insufficient Reason}$ refers to the underlying cause of decision uncertainty, inferred by the MLLM in conjunction with VLM output and contextual information such as environmental cues and anomaly rules. $\texttt{Tools to Use}$ specifies the names of tools used for information enhancement and their corresponding parameters. $\texttt{New Anomaly Rule}$ and $\texttt{New Heuristic Prompt}$ represent the updated anomaly rule and the reformulated heuristic prompt, respectively.

Tool Invocation. PANDA executes the tool functions suggested in the reflection result to enhance both visual and semantic information. The tool invocation process is formulated as:

$$\begin{aligned}\texttt{Result}_{\text{tool\_augmented}} &= \texttt{ToolInvoke}(\texttt{tool}_1, \ldots, \texttt{tool}_n) \\ &= \{\texttt{Text Enhancement Info}, \\ &\quad \texttt{Visual Enhancement Info} = \widehat{c}_t \cup c_s\}.\end{aligned} \tag{9}$$

Here, $\texttt{Text Enhancement Info}$ includes summaries from tool outputs (*e.g.*, detected objects, web search results), while $\texttt{Visual Enhancement Info}$ includes processed video clip $\widehat{c}_t$ and retrieved historical keyframe set $c_s = \{f_1, f_2, ..., f_s\}$.

Refined Reasoning. PANDA updates the reasoning prompt with the newly acquired textual cues:

$$\begin{aligned}\texttt{Prompt}^{\text{refined}}_{\text{reasoning}} &= \texttt{Prompt}_{\text{reasoning}} \cup \{ \\ &\quad \texttt{Text Enhancement Info}, \texttt{New Anomaly Rule}, \\ &\quad \texttt{New Heuristic Prompt}\}\end{aligned} \tag{10}$$

and re-reasoning the enhanced video clip input:

$$\texttt{Result}^{\text{reflection}}_{\text{reasoning}} = \text{VLM}(\widehat{c}_t \cup c_s,\ \texttt{Memory}^{l\text{-steps}}_{\text{visual}},\ \texttt{Prompt}^{\text{refined}}_{\text{reasoning}}). \tag{11}$$

If the returned status is $\texttt{Normal}$ or $\texttt{Abnormal}$, PANDA resumes reasoning at the next timestep. If the status remains $\texttt{Insufficient}$, reflection is re-triggered. To prevent infinite loops, we limit the number of reflection rounds to r. If after r rounds the result is still $\texttt{Insufficient}$, PANDA assigns a default anomaly score corresponding to the $\texttt{Insufficient}$ status and skips the current segment and continues next timestep.

3.4 Self-Improving Chain-of-Memory

To enable PANDA to become increasingly "smarter" over time by accumulating experience through the iterative cycle of reasoning, reflection, and refined reasoning, PANDA equips a self-improving chain-of-memory (CoM) mechanism as shown in Fig. 3. This mechanism enhances both long-term context awareness and consistency in decision-making across video sequences. The CoM comprises two components: short chain-of-memory (short CoM) and long chain-of-memory (long CoM).

Short CoM. In the reasoning stage, short CoM includes both the textual reasoning trace $\texttt{Memory}^{l\text{-steps}}_{\text{text}}$ and its visual counterpart $\texttt{Memory}^{l\text{-steps}}_{\text{visual}}$, as described in subsection 3.2. In the reflection stage, short CoM is represented by the set of past reflection outputs: $\texttt{Result}^{\text{history}}_{\text{reflection}} = \{\texttt{Result}^1_{\text{reflection}}, \texttt{Result}^2_{\text{reflection}}, \ldots, \texttt{Result}^l_{\text{reflection}}\}$.

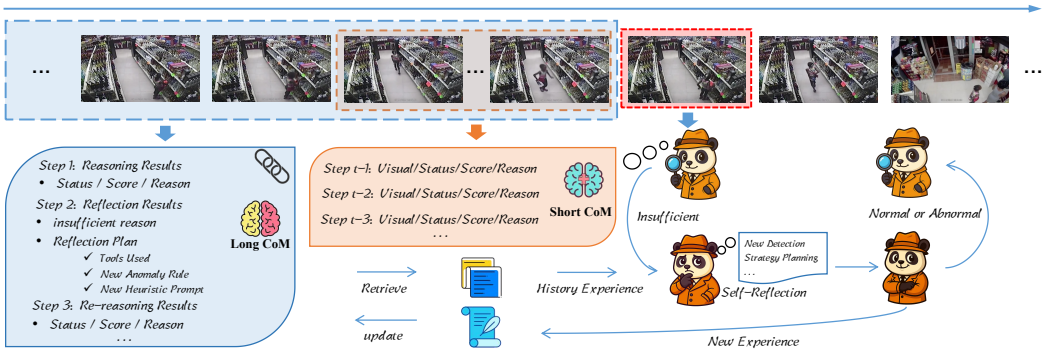

Figure 3: Illustration of Self-Improving Chain-of-Memory.

Long CoM. PANDA also maintains a temporally evolving long CoM: $\texttt{Long-CoM} = \{M_1, M_2, \ldots, M_T\}$, where each memory unit M_t at time step t encapsulates three key outputs: $M_t = \{\texttt{Result}_{\text{reasoning}}, \texttt{Result}_{\text{reflection}}, \texttt{Result}^{\text{refined}}_{\text{reasoning}}\}$. This structure ensures that PANDA retains a complete trace of all decision stages—initial reasoning, reflective analysis, and post-reflection decisions. At the start of a video, LongCoM is empty by design, and PANDA relies on ShortCoM's local window memory for initial reasoning and reflection. As more clips are processed, LongCoM gradually accumulates traces, supporting memory-consistent reasoning and reflection planning. With this self-improving chain-of-memory, PANDA leverages accumulated historical experience to inform both reasoning and reflection, leading to progressively more stable and accurate anomaly detection over time.

4 Experiments

4.1 Experiment Setup

Datasets. We evaluate PANDA on four benchmarks: UCF-Crime [17], XD-Violence [16], UBnormal [34], and CSAD, which represent three distinct settings—multi-scenario (UCF-Crime and XD-Violence), open-set (UBnormal), and complex scenario (CSAD). **UCF-Crime** is a large-scale dataset comprising 1,900 long, untrimmed real-world surveillance videos. It covers 13 types of abnormal events such as fighting, abuse, stealing, arson, robbery, and traffic accidents. The training set includes 800 normal and 810 abnormal videos, while the test set consists of 150 normal and 140 abnormal videos. **XD-Violence** is another large-scale dataset focused on violence detection. It contains 4,754 videos collected from surveillance video, movies, and CCTV sources, encompassing 6 categories of anomaly events. The training and test sets include 3,954 and 800 videos, respectively. **UBnormal** is a synthetic open-set video anomaly detection dataset with a total of 543 videos. It defines 7 categories of normal events and 22 types of anomalies. Notably, 12 anomaly categories in the test set are unseen during training, making it a challenging benchmark for evaluating generalization under open-set conditions. **CSAD** is a complex-scene anomaly detection benchmark constructed in this work. It consists of 100 videos (50 normal and 50 abnormal), sampled from UCF-Crime, XD-Violence, and UBnormal. CSAD includes videos with challenging conditions such as low resolution, poor illumination, high noise levels, and long-range temporal anomalies. It is designed to assess model robustness in complex and degraded environments.

Evaluation Metrics. Following the previous methods [16, 17], we report the Area Under the Curve (AUC) of the frame-level receiver operating characteristic for UCF-Crime, UBnormal, and CSAD. For XD-Violence, we follow the evaluation criterion of average precision (AP) suggested by the work [16] to measure the effectiveness of our method.

Implementation Details. We adopt Langgraph [35] to build the whole agent framework and all experiments are implemented using PyTorch [36] on the A6000 GPU. We use Qwen2.5VL-7B [28] as the VLM for perception and reasoning stages, and Gemini 2.0 Flash [27] as the MLLM for planning and reflection. During the RAG process, the anomaly knowledge base and environment information

Table 1: Comparisons with previous state-of-the-art methods on different datasets. "Expl." stands for "Explanation", indicating whether the output results include interpretations of the detected anomalies. Methods categorized as "Semi", "Weak", or "Instru-Tuned" require training data to adapt to specific scenarios or anomaly types.

Methods	Supervision	Expl.	Manual-free	Mode	Multi-Scenario		Open-Set	Complex Scenario
					UCF (AUC%)	XD (AP%)	UB (AUC%)	CSAD (AUC%)
Specialized methods								
AED-MAE[25] [CVPR2024]	Semi	✗	✗	Offline	-	-	58.50	-
STPAG[38] [CVPR2024]	Semi	✗	✗	Offline	-	-	57.98	-
HL-Net[16] [ECCV2020]	Weak	✗	✗	Offline	82.44	73.67	-	-
RTFM[39] [ICCV2021]	Weak	✗	✗	Offline	84.30	77.81	64.94	-
UR-DMU[40] [AAAI2023]	Weak	✗	✗	Offline	86.97	81.66	59.91	-
VadCLIP[15] [AAAI2024]	Weak	✗	✗	Offline	88.02	84.51	-	-
TPWNG[14] [CVPR2024]	Weak	✗	✗	Offline	87.79	83.68	-	-
VERA[41] [CVPR2025]	Weak	✓	✗	Offline	86.55	70.11	-	64.52
Holmes-VAU[23] [CVPR2025]	Instru-Tuned	✓	✗	Offline	88.96	87.68	56.77	72.47
ZS CLIP[28] [ICML2021]	Training-free	✓	✗	Offline	53.16	17.83	46.2	32.45
LLAVA-1.5[28] [CVPR2024]	Training-free	✓	✗	Offline	72.84	50.26	53.71	47.78
LAVAD [31] [CVPR2024]	Training-free	✓	✗	Offline	80.28	62.01	64.23	57.26
AnomalyRuler [32] [ECCV2024]	Training-free	✓	✗	Offline	-	-	71.90	-
Generalized method								
PANDA (ours)	Training-free	✓	✓	Offline	**84.89**	**70.16**	**75.78**	**73.12**
				Online	82.57	63.57	72.41	71.25

are encoded via the all-MiniLM-L6-v2 model [37], with the knowledge base indexed using FAISS for efficient similarity retrieval. To improve the inference efficiency, the input video is sampled at 1 FPS, and a video clip of $s = 5$ frames is inferred at each time step. PANDA supports both offline and online reasoning modes. In offline reasoning mode, the perception phase is sampling $M = 300$ frames uniformly for the whole video, while only the initial $M = 10$ frames are sampled in online mode. The number of knowledge entries for each type of anomalous event in the anomaly knowledge base is $H = 20$. The maximum number of reflection rounds r is set to 3. The short CoM length $l = 5$ during the reasoning stage. We retrieve the top $k = 5$ anomaly rules from the anomaly knowledge base for each user query. More implementation details, including prompt templates and tool usage, are provided in the supplementary material.

4.2 Comparison with State-of-the-Art Methods

Table 1 compares the performance of PANDA against state-of-the-art specialized VAD methods, including both training-dependent and training-free methods. As shown, PANDA significantly outperforms all existing training-free baselines across all four datasets, even under online settings.

On the UBnormal dataset, which adopts an open-set evaluation protocol where test-time anomalies are unseen during training, PANDA surpasses both training-dependent and training-free approaches. This highlights PANDA's strong generalization capabilities.

PANDA also exhibits notable advantages on CSAD, the complex-scene benchmark introduced in this work, where traditional methods tend to fail under low-quality or temporally extended anomalies. PANDA's superior performance across diverse datasets and conditions demonstrates its robustness and effectiveness as a general-purpose solution for real-world video anomaly detection.

4.3 Analytic Results

Analysis of reflection round r. Table 2(a) shows the effect of varying the number of reflection rounds r on PANDA's performance. We observe that performance improves gradually when increasing r from 1 to 5. Although $r = 5$ yields a slight additional improvement compared to $r = 3$, it introduces more computational overhead due to repeated tool invocation and reasoning steps. To balance efficiency and effectiveness, we adopt $r = 3$ as the default setting in all experiments.

Analysis of rules number k. Table 2(b) analyses the influence of the number of retrieved rules k used during RAG-based anomaly strategy planning. When too few rules are retrieved (*e.g.*, $k = 1$), the system lacks diverse contextual cues to support robust reasoning, resulting in performance degradation.

Table 2: Key hyperparameter analyses on the UCF-Crime dataset.

(a) Analysis of reflection round r.

Reflection Round r	UCF-Crime (AUC%)
1	83.83
3	84.89
5	**84.91**

(b) Analysis of rules number k.

Rules Number k	UCF-Crime (AUC%)
1	82.79
5	**84.89**
9	83.92

(c) Analysis of short CoM length l.

Short CoM length l	UCF-Crime (AUC%)
1	82.92
5	**84.89**
9	84.03

Table 3: Ablation study on the capability of PANDA. **Planning** refers to the self-adaptive scene-aware detection strategy planning, which contains detection planning, adaptive environment perception, and RAG with anomaly knowledge. **Reflection** denotes the tool-augmented self-reflection mechanism. **Memory** corresponds to the chain-of-memory module, encompassing both short-term and long-term components.

Key Capabilities of PANDA			**UCF-Crime (AUC%)**
Planning	**Reflection**	**Memory**	**Performance**
✗	✗	✗	75.25
✓	✗	✗	80.37 (+5.12%)
✓	✓	✗	82.63 (+2.26%)
✓	✓	✓	**84.89 (+2.26%)**

Conversely, setting k too high may introduce noisy or irrelevant rules that dilute reasoning quality. Finally, PANDA achieves optimal performance when setting $k = 5$.

Analysis of short CoM length l. Table 2(c) analyzes the impact of varying the short CoM length l during the reasoning phase. The best performance is achieved when $l = 5$. When the memory length is reduced to $l = 1$, performance drops noticeably due to insufficient temporal information, which limits the model's ability to leverage recent reasoning traces. On the other hand, increasing the memory length to $l = 9$ also leads to performance degradation, likely because excessive memory introduces historical noise that distracts from the current decision-making process.

Ablation Study. Table 3 presents an ablation study examining the contribution of each core capability in PANDA, including self-adaptive scene-aware strategy planning (Planning), tool-augmented self-reflection (Reflection), and chain-of-memory (Memory). The third row serves as the baseline, where PANDA performs direct reasoning solely based on the user-defined query, without planning, reflection, and memory modules. As shown, the performance is relatively poor, with an AUC of 75.25%. Equipping PANDA with the planning capability yields a substantial improvement of +5.12% in AUC. This demonstrates the effectiveness of scene perception, rule retrieval via RAG, and context-aware strategy plans in inspiring the potential of PANDA. Adding the reflection module further improves performance by +2.26%, suggesting that the self-reflection mechanism, enhanced by the integration of external tools, expands PANDA's capability to resolve challenging and ambiguous cases. Finally, incorporating the memory mechanism results in another +2.26% gain, validating the effectiveness of the chain-of-memory design. This module enables PANDA to accumulate experience across time and use it to refine decisions. In summary, each of PANDA's capabilities plays a vital role in enabling generalizable and reliable anomaly detection. The synergistic integration of all modules empowers PANDA as a highly capable agentic AI engineer for generalized VAD. For the ablation study on the key components corresponding to each capability, please refer to the Subsection C.1 in the supplementary material.

4.4 Qualitative Results

Figure 4 shows a visualized example from the UCF-Crime test set to illustrate PANDA's reasoning-reflection process. The left side of the figure shows the anomaly score curve over time. On the right, we visualize PANDA's internal reasoning and reflection process. When the model encounters

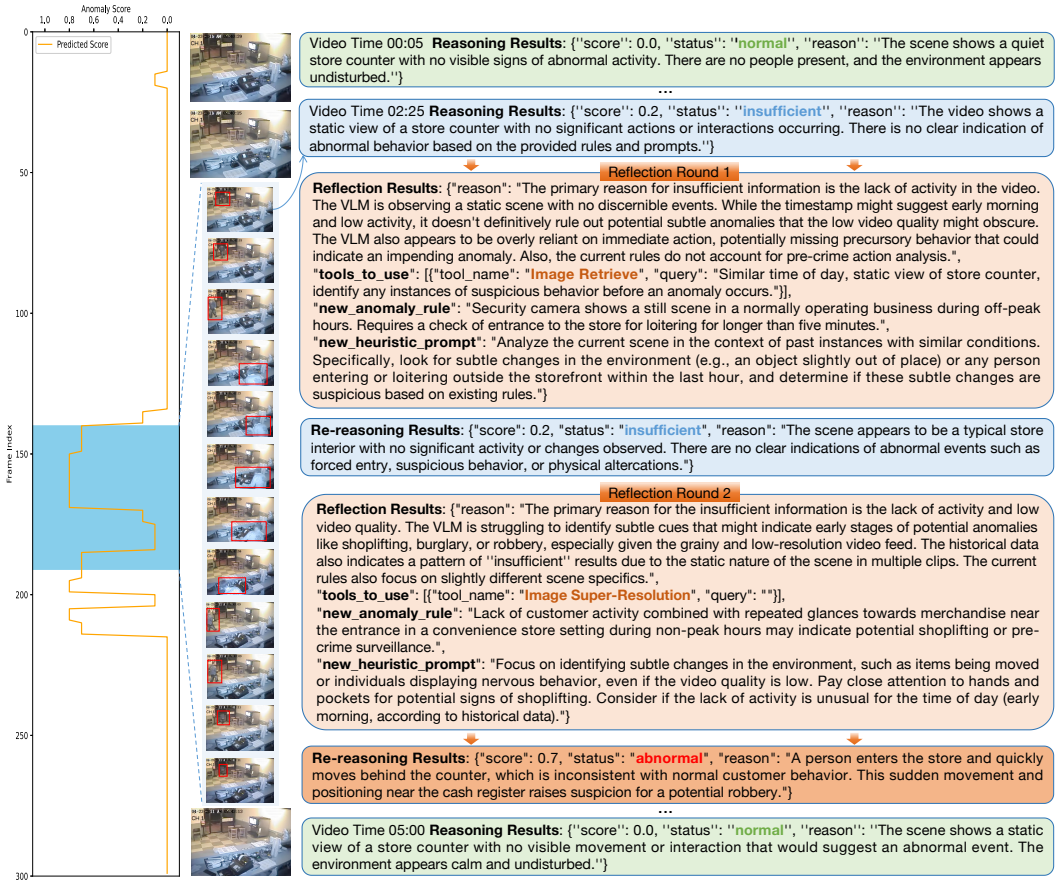

Figure 4: **Visualization of qualitative results on UCF-Crime.** On the left is a visualization of the anomaly score curve. The right side shows the specific reasoning and reflection process of PANDA.

uncertainty and cannot confidently determine whether an anomaly is present, it transitions into the reflection phase. PANDA first analyzes the reason behind the insufficient status from the reasoning stage, and then invokes external tools to acquire complementary information to support decision-making. This case highlights PANDA's capacity for progressive self-refinement and dynamic tool invocation, demonstrating its effectiveness in tackling complex, real-world video anomaly detection scenarios. For more visualization samples, please refer to the Supplementary Material.

5 Conclusion

In this work, we presented PANDA, an agentic AI engineer for generalized VAD that eliminates the need for training data or manually crafted pipelines when faced with various real-world scenarios. PANDA integrates four core capabilities: self-adaptive scene-aware strategy planning, goal-driven heuristic reasoning, tool-augmented self-reflection, and the self-improving chain-of-memory. These capabilities work in concert to enable PANDA to adaptively detect anomalies across diverse, dynamic, and previously unseen environments. Our extensive experiments across multiple benchmarks, including multi-scene, open-set, and complex scenarios, validate PANDA's strong generalization ability and robust performance without any training. These findings highlight PANDA's potential as a generalist VAD solution for real-world scenes.

Acknowledgement

This research is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG3-RP-2022-030).

References

[1] Peng Wu, Chengyu Pan, Yuting Yan, Guansong Pang, Peng Wang, and Yanning Zhang. Deep learning for video anomaly detection: A review. *arXiv preprint arXiv:2409.05383*, 2024.

[2] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *CVPR*, pages 733–742, 2016.

[3] Yannick Benezeth, P-M Jodoin, Venkatesh Saligrama, and Christophe Rosenberger. Abnormal events detection based on spatio-temporal co-occurences. In *CVPR*, pages 2458–2465, 2009.

[4] Yang Cong, Junsong Yuan, and Ji Liu. Sparse reconstruction cost for abnormal event detection. In *CVPR*, pages 3449–3456, 2011.

[5] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *CVPR*, pages 2720–2727, 2013.

[6] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *CVPR*, pages 1975–1981, 2010.

[7] Yu Yao, Xizi Wang, Mingze Xu, Zelin Pu, Yuchen Wang, Ella Atkins, and David J Crandall. Dota: Unsupervised detection of traffic anomaly in driving videos. *IEEE TPAMI*, 45(1):444–459, 2022.

[8] Guansong Pang, Cheng Yan, Chunhua Shen, Anton van den Hengel, and Xiao Bai. Self-trained deep ordinal regression for end-to-end video anomaly detection. In *CVPR*, pages 12173–12182, 2020.

[9] Weixin Luo, Wen Liu, and Shenghua Gao. Remembering history with convolutional lstm for anomaly detection. In *ICME*, pages 439–444, 2017.

[10] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *CVPR*, pages 6536–6545, 2018.

[11] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *CVPR*, pages 1705–1714, 2019.

[12] Zhiwei Yang, Jing Liu, Zhaoyang Wu, Peng Wu, and Xiaotao Liu. Video event restoration based on keyframes for video anomaly detection. In *CVPR*, pages 14592–14601, 2023.

[13] Zhiwei Yang, Peng Wu, Jing Liu, and Xiaotao Liu. Dynamic local aggregation network with adaptive clusterer for anomaly detection. In *ECCV*, pages 404–421, 2022.

[14] Zhiwei Yang, Jing Liu, and Peng Wu. Text prompt with normality guidance for weakly supervised video anomaly detection. In *CVPR*, pages 18899–18908, 2024.

[15] Peng Wu, Xuerong Zhou, Guansong Pang, Lingru Zhou, Qingsen Yan, Peng Wang, and Yanning Zhang. Vadclip: Adapting vision-language models for weakly supervised video anomaly detection. In *AAAI*, volume 38, pages 6074–6082, 2024.

[16] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only look, but also listen: Learning multimodal violence detection under weak supervision. In *ECCV*, pages 322–339, 2020.

[17] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *CVPR*, pages 6479–6488, 2018.

[18] Peng Wu, Xuerong Zhou, Guansong Pang, Zhiwei Yang, Qingsen Yan, Peng Wang, and Yanning Zhang. Weakly supervised video anomaly detection and localization with spatio-temporal prompts. In *ACMMM*, pages 9301–9310, 2024.

[19] Heng Lian, Zhiwei Yang, Zhaoyang Wu, Bo Lin, Tianqiang Huang, and Qiaoru Miao. Vlial: Vision-language instance awareness learning via vlms for weakly supervised video anomaly detection. In *RoboSoft*, pages 23–30, 2025.

[20] Hang Du, Sicheng Zhang, Binzhu Xie, Guoshun Nan, Jiayang Zhang, Junrui Xu, Hangyu Liu, Sicong Leng, Jiangming Liu, Hehe Fan, et al. Uncovering what why and how: A comprehensive benchmark for causation understanding of video anomaly. In *CVPR*, pages 18793–18803, 2024.

[21] Jiaqi Tang, Hao Lu, Ruizheng Wu, Xiaogang Xu, Ke Ma, Cheng Fang, Bin Guo, Jiangbo Lu, Qifeng Chen, and Yingcong Chen. Hawk: Learning to understand open-world video anomalies. *NeurIPS*, 37:139751–139785, 2024.

[22] Huaxin Zhang, Xiaohao Xu, Xiang Wang, Jialong Zuo, Chuchu Han, Xiaonan Huang, Changxin Gao, Yuehuan Wang, and Nong Sang. Holmes-vad: Towards unbiased and explainable video anomaly detection via multi-modal llm. *arXiv preprint arXiv:2406.12235*, 2024.

[23] Huaxin Zhang, Xiaohao Xu, Xiang Wang, Jialong Zuo, Xiaonan Huang, Changxin Gao, Shanjun Zhang, Li Yu, and Nong Sang. Holmes-vau: Towards long-term video anomaly understanding at any granularity. *arXiv preprint arXiv:2412.06171*, 2024.

[24] Zhiwei Yang, Chen Gao, Jing Liu, Peng Wu, Guansong Pang, and Mike Zheng Shou. Assistpda: An online video surveillance assistant for video anomaly prediction, detection, and analysis. *arXiv preprint arXiv:2503.21904*, 2025.

[25] Nicolae-C Ristea, Florinel-Alin Croitoru, Radu Tudor Ionescu, Marius Popescu, Fahad Shahbaz Khan, Mubarak Shah, et al. Self-distilled masked auto-encoders are efficient video anomaly detectors. In *CVPR*, pages 15984–15995, 2024.

[26] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[27] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[28] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[29] OpenGVLab Team. Internvl2: Better than the best—expanding performance boundaries of open-source multimodal models with the progressive scaling strategy, 2024.

[30] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[31] Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. Harnessing large language models for training-free video anomaly detection. In *CVPR*, pages 18527–18536, 2024.

[32] Yuchen Yang, Kwonjoon Lee, Behzad Dariush, Yinzhi Cao, and Shao-Yuan Lo. Follow the rules: reasoning for video anomaly detection with large language models. In *ECCV*, pages 304–322. Springer, 2024.

[33] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *NeurIPS*, 33:9459–9474, 2020.

[34] Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. Ubnormal: New benchmark for supervised open-set video anomaly detection. In *CVPR*, pages 20143–20153, 2022.

[35] AI LANGCHAIN. Langgraph: Multi-agent framework for llms. 2025. *URL https://github. com/langchain-ai/langgraph. Accessed*, pages 01–24, 2025.

[36] Sagar Imambi, Kolla Bhanu Prakash, and GR Kanagachidambaresan. Pytorch. *Programming with TensorFlow: solution for edge computing applications*, pages 87–104, 2021.

[37] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *EMNLP*, 2020.

[38] Ayush K Rai, Tarun Krishna, Feiyan Hu, Alexandru Drimbarean, Kevin McGuinness, Alan F Smeaton, and Noel E O'connor. Video anomaly detection via spatio-temporal pseudo-anomaly generation: A unified approach. In *CVPR*, pages 3887–3899, 2024.

[39] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In *ICCV*, pages 4975–4986, 2021.

[40] Hang Zhou, Junqing Yu, and Wei Yang. Dual memory units with uncertainty regulation for weakly supervised video anomaly detection. In *AAAI*, volume 37, pages 3769–3777, 2023.

[41] Muchao Ye, Weiyang Liu, and Pan He. Vera: Explainable video anomaly detection via verbalized learning of vision-language models. In *CVPR*, pages 8679–8688, June 2025.

[42] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *CVPR*, pages 16901–16911, 2024.

[43] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *ICCV*, pages 1905–1914, 2021.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021.

[45] Tavily. Tavily search api. `https://www.tavily.com`.

Technical Appendices and Supplementary Material

A Overview of Technical Appendices and Supplementary Material

This technical appendices and supplementary material provide additional information not included in the main paper. Specifically:

- **Section B** offers further clarification on methodological and experimental details.
- **Section C** presents additional ablation studies and parameter analysis experiments.
- **Section D** details the toolset employed by PANDA.
- **Section E** shows the prompts used at each stage of the PANDA framework.
- **Section F** provides additional visualizations of qualitative results across various datasets.
- **Section G** discusses the current limitations, broader impacts, and future work directions.

B Additional Methodological and Experimental Details

B.1 Visualization of the PANDA Detailed Pipeline

Figure 5 presents a visualized case study on a test example from the UCF-Crime dataset, illustrating the detailed execution process of PANDA across its core components: anomaly knowledge base construction, self-adaptive scene-aware strategy planning, goal-driven heuristic reasoning, tool-augmented self-reflection, and the self-improving chain-of-memory mechanism.

B.2 Clarification on Evaluation Modes and SOTA Comparison

In Table 1 of the main paper, we distinguish between *offline* and *online* settings based on whether future information is utilized when reasoning over a given frame or clip. If future information is accessed, the method is considered offline; otherwise, it falls under the online setting. For offline evaluation, we follow the SOTA methods AED-MAE, LAVAD, and AnomalyRuler, which are compared in Table 1 of the main paper, and apply temporal smoothing (mean filtering, window size=10) on the final anomaly scores.

Among the compared methods, Holmes-VAU is a fine-tuned VLM-based approach that leverages detailed anomaly annotations via instruction tuning. It was originally evaluated only on UCF-Crime and XD-Violence. Its results on UBnormal and CSAD are reproduced by us, without any re-training, using its publicly released model. As seen in Table 1 of the main paper, PANDA significantly outperforms Holmes-VAU on UBnormal, an open-set dataset that includes unseen anomaly types. While Holmes-VAU slightly surpasses PANDA in online mode on CSAD, this is primarily due to CSAD including a large number of videos derived from UCF-Crime and XD-Violence—the original training sets of Holmes-VAU. Notably, PANDA still achieves superior performance under the offline setting, underscoring its strong generalization capability. These results demonstrate the limitations of relying solely on fine-tuned VLMs when facing domain shift and complex real-world conditions. Additionally, we report new evaluation results for three prominent training-free baselines: ZS-CLIP, LLaVA-1.5, and LAVAD, on UBnormal and CSAD. PANDA consistently and substantially outperforms all of them across both datasets. Together, these experimental results reinforce the strength of PANDA as a generalist, fully automated VAD agent, capable of adapting its reasoning to scene-specific conditions without supervision or hand-crafted engineering.

C Additional Experiments

C.1 Ablation of Key Component

Table 4 presents the results of an ablation study evaluating the contribution of PANDA's six core components. The third row in the table corresponds to a baseline that directly queries the VLM using only the user-defined anomaly description without leveraging any PANDA modules. This baseline yields notably poor performance, confirming that naive prompting alone is insufficient. As more modules are incrementally added—namely, detection strategy planning, adaptive scene perception, RAG with anomaly knowledge, self-reflection, short CoM, long CoM—the performance steadily improves. When all six components are combined, PANDA achieves its best overall performance. These results demonstrate that each individual module contributes positively to the final performance and validates the effectiveness of our whole PANDA framework design.

Table 4: Ablation results of PANDA component.

Key Components of PANDA						Dataset
Detection Strategy Planning	Self-Adaption Scene-Aware	RAG with Anomaly Knowledge	Self-Reflection	Short CoM	Long CoM	UCF-Crime(AUC%)
✗	✗	✗	✗	✗	✗	75.25
✓	✗	✗	✗	✗	✗	77.01
✓	✓	✗	✗	✗	✗	78.92
✓	✓	✓	✗	✗	✗	80.37
✓	✓	✓	✓	✗	✗	82.63
✓	✓	✓	✓	✓	✗	83.94
✓	✓	✓	✓	✓	✓	**84.89**

Table 5: Additional Experiments.

(a) Impact of different MLLMs.

Different MLLMs	UCF-Crime (AUC%)
Qwen2.5-72B	84.03
DeepseekV3	84.72
GPT4o	**84.97**
Gemini 2 Flash	84.89

(b) Effect of input clip length s.

Input Clip Length s (Number of frames)	UCF-Crime (AUC%)
1	84.25
3	84.56
5	**84.89**
7	83.15

(c) Analysis of Inference speed.

Datasets Name	Average speed of inference (FPS)
UCF-Crime	0.82
XD-Violence	0.86
UBnormal	0.79
CSAD	0.53

C.2 Impact of Different MLLMs

Table 5a compares the performance of PANDA when integrated with different multi-modal large language models (MLLMs). GPT-4o and Gemini 2 Flash represent proprietary models, while DeepSeek-V3 and Qwen2.5-72B are open-source alternatives. As shown, GPT-4o achieves the highest performance. However, we adopt Gemini 2 Flash in our main pipeline due to its strong trade-off between performance and cost-effectiveness.

Notably, although Qwen2.5-72B yields the lowest performance among the compared models, it remains significantly superior to prior training-free baselines. Given its open-source nature and ease of local deployment, it serves as a practical and scalable option for resource-constrained scenarios.

C.3 Effect of Input Clip Length

Table 5b analyzes how varying the number of frames in each input video clip affects PANDA's performance. As the input length increases from 1 to 5 frames, detection accuracy steadily improves, suggesting that short-range temporal cues are beneficial to the reasoning process. However, when the clip length is extended to 7 frames, performance noticeably drops. We hypothesize that this is due to the binary labeling strategy used during evaluation—if a clip is anomalous, all frames of the clip are scored as anomalous. For longer clips that may contain both normal and abnormal frames, this scoring scheme introduces noise, leading to performance degradation.

C.4 Analysis of Inference Speed

Table 5c reports the average inference speed of PANDA across different datasets. As observed, PANDA achieves similar inference times on UCF-Crime, XD-Violence, and UBnormal. However, a noticeable slowdown is observed on the CSAD dataset. This is primarily because CSAD contains videos with complex conditions and scenarios, leading PANDA to more frequently enter the reflection stage. Since the reflection stage involves invoking additional tools, it introduces greater computational overhead. Despite this, the overall average inference speed of PANDA remains acceptable for non-time-sensitive applications, demonstrating its practical feasibility in real-world deployments where latency is not a critical constraint.

D Toolset Details

PANDA is equipped with a modular and extensible set of tools designed to enhance video content analysis, mitigate visual degradation, and provide external contextual information. These tools are automatically selected and invoked during the self-reflection stage based on the detection context. Below, we provide a detailed summary of each tool integrated into the PANDA framework.

Object Detection. PANDA employs the YOLOWorld [42] model pretrained on a wide set of open-world concepts. It supports fine-grained category-specific detection including actions like "person hitting another", "person setting something on fire", and "person stealing". This enables robust anomaly-related scene understanding through bounding-box localization and category labels.

Image Denoising. PANDA uses OpenCV's fast non-local means filter for image denoising. It reduces color and spatial noise in frames using adaptive filtering, helping enhance clarity in low-light or noisy environments.

Image Deblurring. PANDA applies unsharp masking with Gaussian blur subtraction to sharpen edge details for motion blur or out-of-focus issues. This lightweight enhancement improves perceptual clarity without the need for retraining.

Image Brightness Enhancement. PANDA uses OpenCV's CLAHE (Contrast Limited Adaptive Histogram Equalization) on the L-channel in LAB color space. This ensures localized brightness normalization for dimly lit or overexposed frames.

Image Super-Resolution. PANDA integrates the Real-ESRGAN [43] model for resolution enhancement. It improves detail preservation and restores textures in low-resolution videos using a deep RRDBNet-based super-resolution pipeline.

Image Retrieval. PANDA uses CLIP-based [44] visual-textual retrieval to match current queries (e.g., "robbery incident") with previously seen keyframes. Cosine similarity between CLIP embeddings is used for scoring relevance.

Web Search. PANDA leverages the Tavily Search API [45] for querying web content related to unknown or uncertain anomalies. Search results are parsed into structured summaries that can be referenced in the reasoning process.

Image Zooming. PANDA leverages a bicubic interpolation-based zooming tool to magnify regions that require enhanced spatial detail using a specified zoom factor. This is useful when detecting small-scale interactions or distant activities.

All tools are dynamically invoked during the reflection stage via the MLLM-generated reflection plan. Each tool outputs enhanced frame sets and structured summaries that are used to augment the reasoning prompt for follow-up reasoning steps.

E System Prompts

In this section, we present the detailed prompts used by PANDA across its core stages. Figure 6 illustrates the prompt used during anomaly knowledge base construction. Figure 7 shows the self-adaptive environmental perception prompt. Figure 8 presents the prompt for anomaly detection strategy planning. Figure 9 demonstrates the goal-driven heuristic reasoning prompt. Figure 10 displays the prompt used during the tool-augmented self-reflection phase.

F Additional Visualization Results

Figures 11 and 12 further show the qualitative results of the samples on the XD-Violence and UBnormal test sets.

G Discussions

G.1 Limitations

PANDA currently integrates a curated set of commonly used tools for enhancement and reasoning. While these tools suffice for most general scenarios, expanding the toolkit to accommodate domain-specific modalities (e.g., thermal imaging) could broaden PANDA's applicability.

G.2 Broader Impacts

PANDA advances the paradigm of automated video anomaly detection by integrating vision-language models and decision-time tool augmentation into a unified AI agent framework. This work has the potential to improve the robustness and interpretability of security monitoring systems, enabling adaptive deployment across diverse environments without training. Moreover, We also recognize the importance of ethical considerations: systems like PANDA must be deployed responsibly, with attention to privacy protection, fairness, and minimizing unintended surveillance harms.

G.3 Future Work

Enhancing Real-Time Adaptability. To make PANDA more suitable for practical deployments, future work could focus on reducing inference latency and optimizing tool invocation paths.

Improving Spatial Reasoning and Localization. Currently, PANDA focuses on frame-level or clip-level anomaly identification. Incorporating spatial anomaly localization, such as identifying the precise region or object involved in an abnormal event, could significantly expand its utility in surveillance systems.

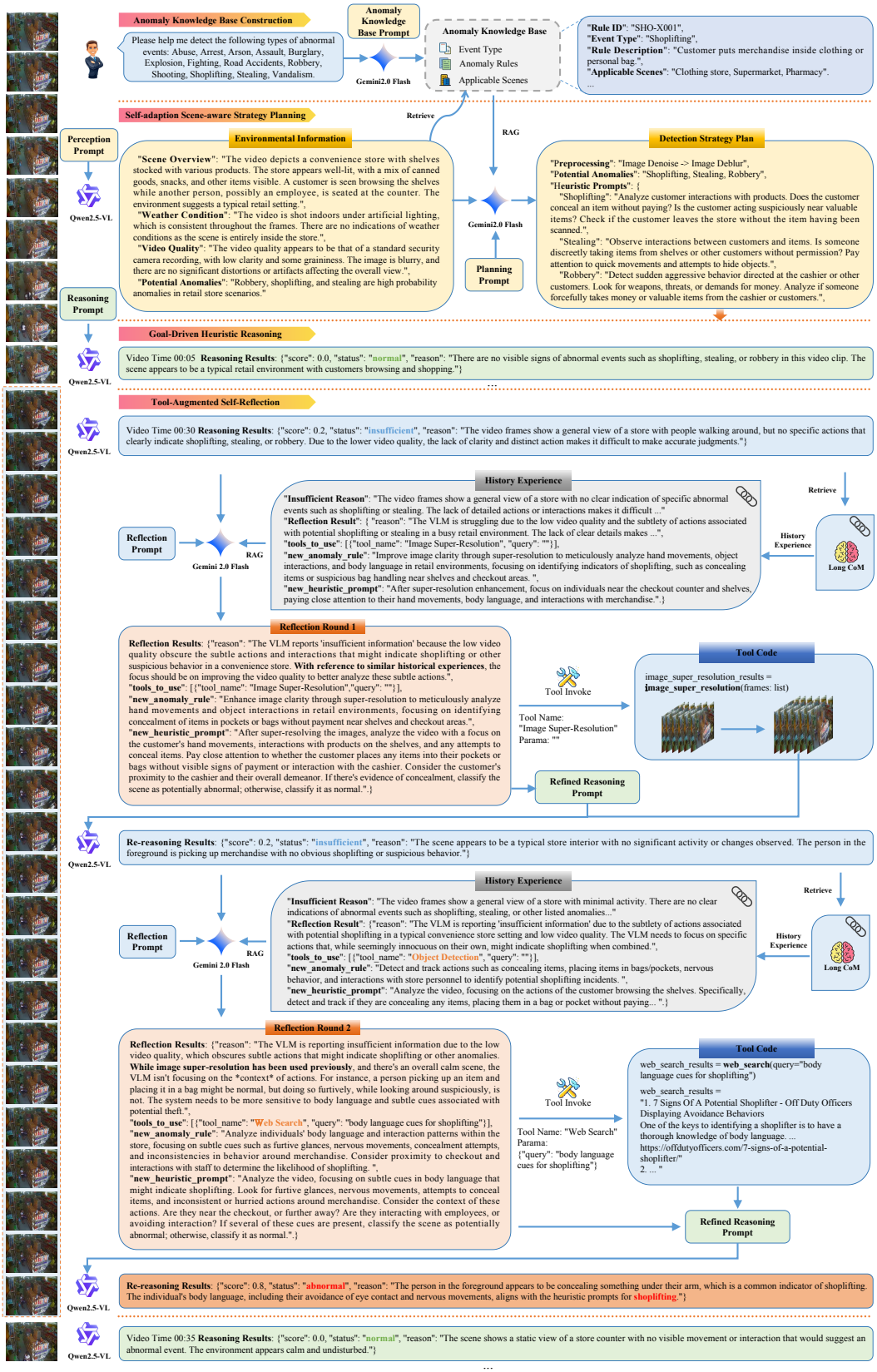

Figure 5: **Visualization of the PANDA detailed pipeline.** We show a more detailed pipeline for PANDA by visualizing the results of a test sample on UCF-Crime.

Anomaly Knowledge Base Construction Prompt

```
You are an expert in designing detection rules for video anomaly detection. Based
on the user's specified types of abnormal events, your task is to generate 20
comprehensive and diverse detection rules for each event type.

User Requirement:
{user_query}

Each rule must include the following four fields:
1. Rule ID: A unique identifier for the rule (e.g., "FIG-X001").
2. Event Type: The corresponding abnormal event category (e.g., "Fighting").
3. Rule Description: A concise and clear description of the behavioral pattern
that defines this event.
4. Applicable Scenes: A list of real-world scenarios where this rule may be
applicable (e.g., "Street, Shopping mall, School playground").
The rules should cover a wide range of realistic situations for each event type.

Please output your response in the following structured JSON format:
[{
  "Rule ID": "FIG-X001",
  "Event Type": "Fighting",
  "Rule Description": "Pulling hair or grabbing clothes during struggle",
  "Applicable Scenes": "Shopping mall, Playground, Street corner"
},
...
]
```

Figure 6: System prompt for anomaly knowledge base construction.

Self-Adaption Environmental Perception Prompt

```
You are an expert in video anomaly perception. Your task is to perform an initial
understanding and analysis of the provided video frames based on the user's
specified requirements.

User Requirement:
{user_query}

Please respond by completing the following four aspects:
1. Scene Overview: Describe the environment shown in the video (e.g., shopping
mall, office, street, surveillance corridor) and briefly summarize the main
activities or events observed.
2. Weather Condition: Describe the visual lighting conditions (e.g.,
daytime/nighttime, sunny/overcast, bright/dim).
3. Video Quality:  Comment on the overall quality of the video (e.g., clear,
blurry, noisy, low frame rate, low light.etc.).
4. Potential Anomalies: Based on the visual content, what types of abnormal events
are likely to occur in this video (e.g., Fighting, Stealing).

Please return your output strictly in the following JSON format:
 {
  "Scene Overview": "...",
  "Weather Condition": "...",
  "Video Quality": "...",
  "Potential Anomalies": "..."
}
```

Figure 7: System prompt for self-adaptive environmental perception.

Anomaly Detection Strategy Planning Prompt

```
You are a strategic expert in video anomaly detection, specializing in planning
effective detection strategies based on user-defined requirements, video
environment information, and scene-specific anomaly rules. Your role is to
generate an optimal plan that guides the analysis module in accurately detecting
anomalies in the current video scene.
User Requirement:
{user_query}

Video Environment Information:
1. Scene Overview: {env_info.get("Scene Overview", "Unknown")}
2. Weather Condition: {env_info.get("Weather Condition", "Unknown")}
3. Video Quality: {env_info.get("Video Quality", "Unknown")}
4. Potential Anomalies: {env_info.get("Potential Anomalies", "Unknown")}

Anomaly Detection Rules:
{anomaly_rules}

Based on the user's requirement, the preliminary video environment information,
and the provided anomaly rules, please design a strategy tailored for this video
scenario. Your response must include the following three components:
1. Preprocessing Recommendations and Pipeline
    Suggest a sequence of preprocessing steps (e.g., Image Deblurring, Brightness
Enhancement, Image Denoising) that can help improve video quality and support
better anomaly detection, especially if the video is of poor quality.

2. Potential Anomaly Types
    Based on the preliminary video environment information and the given scene-
related anomaly rules, further infer and list the most possible types of anomalies
in this scenario.

3. Heuristic Prompts for VLM
    Using the anomaly rules as guidance, craft chain-of-thought-style heuristic
prompts for each potential anomaly type. These prompts are intended to assist
Visual Language Model in performing accurate anomaly judgments.

Please return your output strictly in the following JSON format:
{
  "Preprocessing": "Step1 -> Step2 -> ...",
  "Potential Anomalies": "Fighting, Stealing, ...",
  "Heuristic Prompts": {
    "Fighting": "Heuristic prompt with reasoning steps...",
    "Stealing": "Heuristic prompt with reasoning steps...",
    ...
  }
}
Example Output:
{
  "Preprocessing": "Image Brightness Enhancement -> Image Denoising",
  "Potential Anomalies": "Fighting, Stealing",
  "Heuristic Prompts": {
    "Fighting": "Observe the number of people, their movements, and interactions.
If two individuals are repeatedly making aggressive contact, consider it a
potential fight.",
    "Stealing": "Identify solitary individuals interacting with objects,
especially if they conceal items or leave quickly without paying."
  }
}
```

Figure 8: System prompt for anomaly detection strategy planning.

Goal-Driven Heuristic Reasoning Prompt

```
You are a highly skilled expert in video anomaly detection, specializing in
identifying abnormal events through temporal and spatial analysis of visual
evidence.
Given a sequence of video frames, user requirements, potential anomalies, anomaly
detection rules, heuristic prompts, and enhancement/reflection information, your
task is to assess the likelihood of abnormal events in the current video clip.

You must output:
- A soft anomaly score between **0.0 (clearly normal)** and **1.0 (clearly
abnormal)**.
- A status label from: **"normal"**, **"abnormal"**, or **"insufficient"**.
- A reason justifying your decision.

User Requirement:
{user_query}

Historical Detection Info:
{history_result_prompt if history_result_prompt.strip() else "No reliable
historical detection information available."}

Current Video Clip Index:
Clip {index}

Potential Anomalies:
{planning_info.get('potential_anomalies', '')}

Anomaly Detection Rules:
{anomaly_rules}

Heuristic Prompt:
{planning_info.get('heuristic_prompts', '')}

Enhancement and Reflection Information:
{formatted_enhancement_prompt}

Your analysis should follow three steps:
1. Describe the main visible actions and interactions between people or objects in
the scene.
2. Assess how strongly these actions match any known abnormal event patterns using
the provided rules and prompts. The provided anomaly rules may not be
comprehensive, so you also apply your own expert reasoning.
3. Based on your assessment, assign a score and label, and explain your reasoning
clearly.
4. If the [Enhancement and Reflection Information] section provides additional
information, you should refer to it emphatically.

Scoring Guidelines:
- A score close to **1.0** indicates clear and confident abnormal behavior.
- A score close to **0.0** indicates clearly normal behavior.
- A score near **0.5** means uncertain, ambiguous behavior or mixed signals.

Examples of valid reasons for "insufficient":
- "The entire scene is too blurry or dark, making it difficult to distinguish any
actions."
- "All persons are either occluded or out of frame."
- "Only partial limbs are visible and motion cues are unclear."

Please strictly output your response in the following JSON format:
{
      "score": float,  //anomaly score in [0.0, 1.0]
      "status": "normal/abnormal/insufficient",
      "reason": "A detailed explanation of your reasoning..."
    }
```

Figure 9: System prompt for goal-driven heuristic reasoning.

Tool-Augmented Self-Reflection Prompt

```
You are a reflection assistant within a video anomaly detection system.
The current VLM analysis module has returned "insufficient statu" for determining
whether an abnormal event occurred in the given video clip.
Your task is to critically analyze the situation based on the provided context and
recommend solutions.

Here is the contextual information:
- User Requirement: {user_query}
- Video Environment Information:
    **Scene Overview: {env_info.get("Scene Overview", "Unknown")}
    **Weather Condition: {env_info.get("Weather Condition", "Unknown")}
    **Video Quality: {env_info.get("Video Quality", "Unknown")}
- Anomaly Detection Rules:
{anomaly_rules}
- Potential Anomalies: {planning_info.get('Potential Anomalies', '')}
- Historical Detection Results:
{historical_results}
- Current VLM Output Reason: {reason}
- Information Enhancement tools Already Used: {tools_already_used}
- History Experience: {memory_context}

Based on this information, your tasks are:
1. Analyze and determine the primary reasons for the insufficient information.
2. Recommend which tools from the available options should be used to enhance the
information for better anomaly detection.
{tool_description_text}
3. For any selected tool that requires a 'query' input (e.g., image_retrieve,
web_search), generate an appropriate query based on the context; otherwise leave
the 'query' field empty.
4. Propose a new representative anomaly detection rule derived from the current
situation to better support future VLM analysis.
5. Propose an additional heuristic prompt based on the context and your analysis
to better guide the VLM toward an accurate judgment.

Please output your response in the following structured JSON format:
 {
      "reason": "...your analysis of why the information is insufficient...",
      "tools_to_use": [
        {
          "tool_name": "One of the most critical tools.",
          "query": "generated query if needed, otherwise leave empty"
        }
      ],
      },
      "new_anomaly_rule": "...a new representative anomaly rule derived from your
analysis and context...",
      "new_heuristic_prompt": "...additional guidance to help the VLM make a more
accurate judgment..."
          }
Important Notes:
- When calling tools, make sure you don't duplicate any of the information
enhancement tools that have already been applied, and use only one of the most
critical tools at a time.
- If all available information enhancement tools have been exhausted, you should
directly suggest in the 'new_heuristic_prompt' how to guide the VLM analysis
module to make the most reasonable judgment based on incomplete evidence.
- If the current context provides enough information to make a clear judgment,
please directly guide the VLM analysis module in the 'new_heuristic_prompt' to
conclude whether the event is abnormal or normal.
```

Figure 10: System prompt for tool-augmented self-reflection.

Figure 11: Visualization of qualitative results for a sample on the XD-Violence test set.

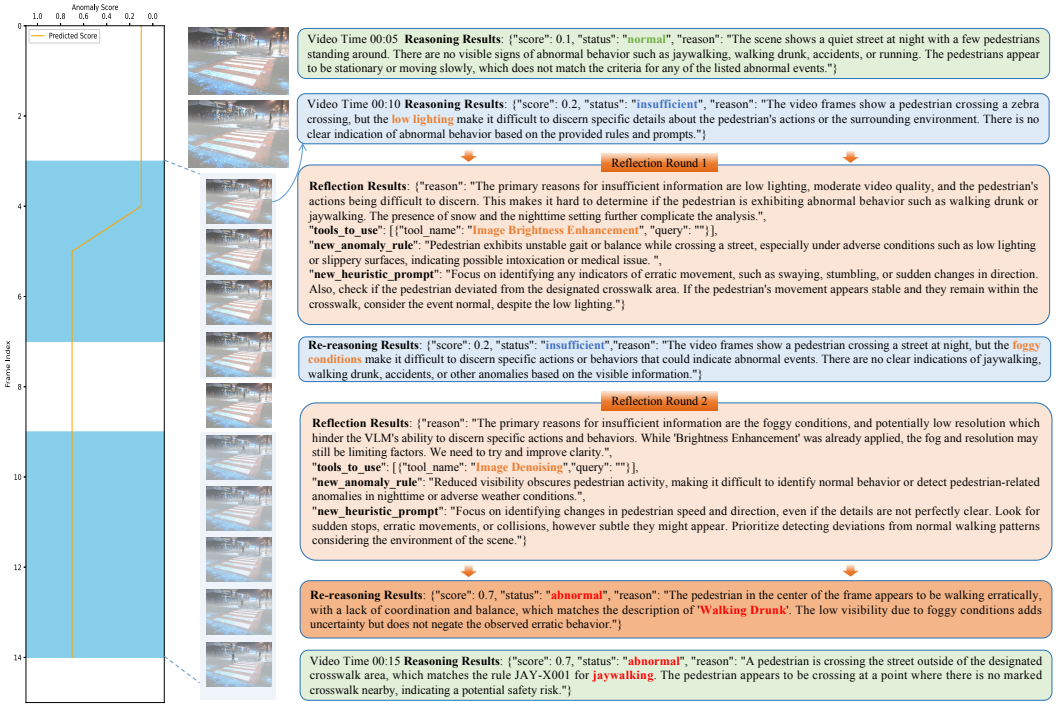

Figure 12: Visualization of qualitative results for a sample on the UBnormal test set.

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 abstract and introduction of the paper describe in detail the contribution and scope of the research.

   Guidelines:

   - The answer NA 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 NA 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 describe the limitations of the method in the supplementary material.

   Guidelines:

   - The answer NA 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: [Yes]

Justification: The methodology section 3 of the paper describes in detail the implementation process and details of the entire methodological framework.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The experimental part of the paper details the experimental environment platform on which the method is executed, as well as the specific parameters of each module of the method, which fully ensures the reproducibility of the experimental results.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: The dataset we used is publicly available, and we will share our codes and database publicly once the paper is accepted.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.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://nips.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, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The implementation details section 4.1 of the paper details the experimental implementation details and its parameter settings.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: Our paper performs deterministic evaluations on the fixed dataset. In the case of fixed random seeds, our experiments perform deterministic results.

Guidelines:

- The answer NA 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: Details of the execution of the experiment and the platform on which it was run are described in the experimental setup of the main paper and in the supplementary material.

   Guidelines:

   - The answer NA 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: The research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Broader impacts are described in detail in the supplementary material.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not contain such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: The creators or original owners of assets used in the paper are properly credited.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: No new assets are introduced in the paper.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: The paper does not include crowdsourcing experiments and research with human subjects.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: The paper does not contain experiments IRB required.

    Guidelines:

    - The answer NA 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 rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Our method uses VLM and MLLM for inference, and the type of LLM used is described in the experimental setup subsection 4.1.

    Guidelines:

    - The answer NA 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 (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.