PROTOTYPE-DRIVEN HIERARCHICAL RULE INDUCTION AND MATCHING FOR EFFICIENT VIDEO ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video anomaly detection (VAD) aims to identify rare and unexpected events in video. Though training-free[1] VAD using vision-language models (VLMs) has shown promise, existing pipelines rely on dense-frame semantic generation and reasoning, which is computationally expensive and not good at modeling context-dependent anomalies. To address these issues, we propose **ProtoRule-VAD**, an efficient framework that converts normal videos into reusable hierarchical semantic rules without anomalous supervision or model parameter updates. In the offline stage, scene-level and object-level normal prototypes are constructed from normal videos and used to induce contrastive hierarchical rules with a pretrained VLM, including scene-level, object-level, and condition-level rules. During inference, motion saliency and prototype-based visual novelty are combined to prioritize candidate segments. Condition-level rules then select relevant scene-level and object-level rules for selective rule matching and anomaly scoring.ProtoRule-VAD delivers competitive performance and interpretable rule-based evidence without requiring anomalous data and fine-tuning. Notably, our processing frame rate is over 500 FPS on one NVIDIA A6000 GPU, substantially improving online inference throughput over existing VLM-based VAD methods.

1 INTRODUCTION

Video anomaly detection (VAD) (Alinezhad Noghre et al., 2025) aims to identify rare and unexpected events in long surveillance video streams (Liu et al., 2025), playing an indispensable role in real-world applications ranging from public security to autonomous driving. The core challenge in VAD is that anomalies are inherently rare, diverse, and long-tailed, making it exceedingly difficult to collect representative anomaly data at scale. This challenge has motivated two dominant learning paradigms. One-class methods (Hasan et al., 2016; Luo et al., 2017; Gong et al., 2019b; Ruff et al., 2018; Liu et al., 2018) learn normal patterns from only normal videos and detect deviations through reconstruction errors, prediction errors, or learned feature distributions. Weakly supervised methods (Feng et al., 2021; Chen et al., 2023) exploit video-level anomaly labels and formulate VAD as a multiple instance learning problem to learn discriminative anomaly representations. Despite their strong detection performance, both paradigms rely heavily on task-specific training and output opaque anomaly scores, offering no semantic rationale behind their decisions. This black-box nature inherently limits their interpretability and interaction with real-world applications.

The advent of large vision-language models (VLMs) and large language models (MLLMs) has sparked a paradigm shift toward open-vocabulary, interpretable VAD. While existing methods (Wu et al., 2024; Zhang et al., 2025) attempt to adapt these foundation models via task-specific fine-tuning, the exorbitant training costs have driven a growing interest in leveraging *frozen* VLMs through prompt engineering or language-based reasoning. To elicit the reasoning capabilities of frozen models, VERA (Ye et al., 2025) introduces verbalized learning by optimizing guiding questions with an optimizer VLM, enabling interpretable anomaly reasoning without modifying the underlying model parameters.

[1]Following existing works(Li et al., 2025; Pei et al., 2026; Zanella et al., 2024b), training-free in this paper means no fine-tuning of the Multimodal Large Language Model.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Nevertheless, anomalous training data may not be available and the multi-round verbal optimization incurs a heavy computational burden prior to deployment.

To reduce dependence on anomalous data, recent works investigate how to derive semantic knowledge from normal observations. AnomalyRuler (Yang et al., 2024a) induces normality rules from a small number of normal reference samples and performs rule-based anomaly discrimination. LAVAD (Zanella et al., 2024b) depend on generating dense captions for every video frame to conduct semantic anomaly analysis. Although these approaches enable semantic reasoning without anomaly supervision, they face a trade-off between semantic coverage and inference efficiency. Dense caption generation or repeated language-model reasoning over long videos introduces substantial computational overhead and may ignore the contextual relationships among scenes, objects, and behaviors.

Meanwhile, real-world anomalies are often context-dependent (Sun et al., 2020; Tang et al., 2024). The same behavior can be normal in one scenario (e.g., running on a sports field) but abnormal in another (e.g., running in a shopping mall). Therefore, effective anomaly reasoning requires not only identifying unusual visual patterns but also understanding the compatibility between scene context, objects, and behaviors. Existing training-free approaches have explored different forms of semantic reasoning, such as event-level modeling (Shao et al., 2025), hierarchical temporal reasoning (Li et al., 2025), and customizable anomaly descriptions (Ahn et al., 2026). However, explicitly organizing target-domain normal knowledge into structured semantic representations remains underexplored.

We target a practical setting of anomaly detection with no anomalous supervision or model parameter updates, while aiming for efficient inference and context-aware semantic evidence. Our key insight is that expensive semantic knowledge construction can be performed offline from compact normal representations, while online inference can focus on selective semantic verification. Instead of repeatedly reasoning over dense video frames, we first summarize normal videos into representative prototypes and induce a hierarchical rule bank describing scene patterns, object behaviors, and scene-object conditions. During inference, lightweight visual cues prioritize candidate segments, and semantic rule matching is selectively applied only to these candidates.

To this end, we propose ProtoRule-VAD, a prototype-driven hierarchical rule induction framework for efficient video anomaly detection. In the offline stage, scene-level and object-level prototypes are constructed from normal training videos through feature clustering, and a pretrained VLM is prompted to induce contrastive rules describing normal patterns and potential violations. These rules form a hierarchical rule bank consisting of scene-level, object-level, and condition-level rules. In the online stage, motion saliency and prototype-based visual novelty are combined to prioritize candidate segments, while condition-level rules guide selective matching with relevant scene-level and object-level rules for anomaly scoring.

Our main contributions are as follows:

- We propose a prototype-driven hierarchical rule induction framework for efficient VAD, which compresses normal videos into structured scene-, object-, and condition-level rules using a pretrained VLM without anomalous supervision or VLM fine-tuning.
- We design a selective hierarchical rule matching strategy, where lightweight visual cues prioritize candidate segments and condition-level rules provide context-aware routing for scene-level and object-level semantic verification, reducing unnecessary semantic computation during inference.
- We conduct extensive experiments on UCF-Crime, XD-Violence, and ShanghaiTech, demonstrating that ProtoRule-VAD achieves competitive anomaly detection performance among training-free approaches while providing substantially higher online inference throughput.

2 Related Work

Video anomaly detection.

Video anomaly detection. Traditional video anomaly detection methods mainly follow two paradigms: normality modeling and weakly supervised discriminative learning. Normality modeling learns regular patterns from normal videos and detects anomalies through reconstruction or prediction

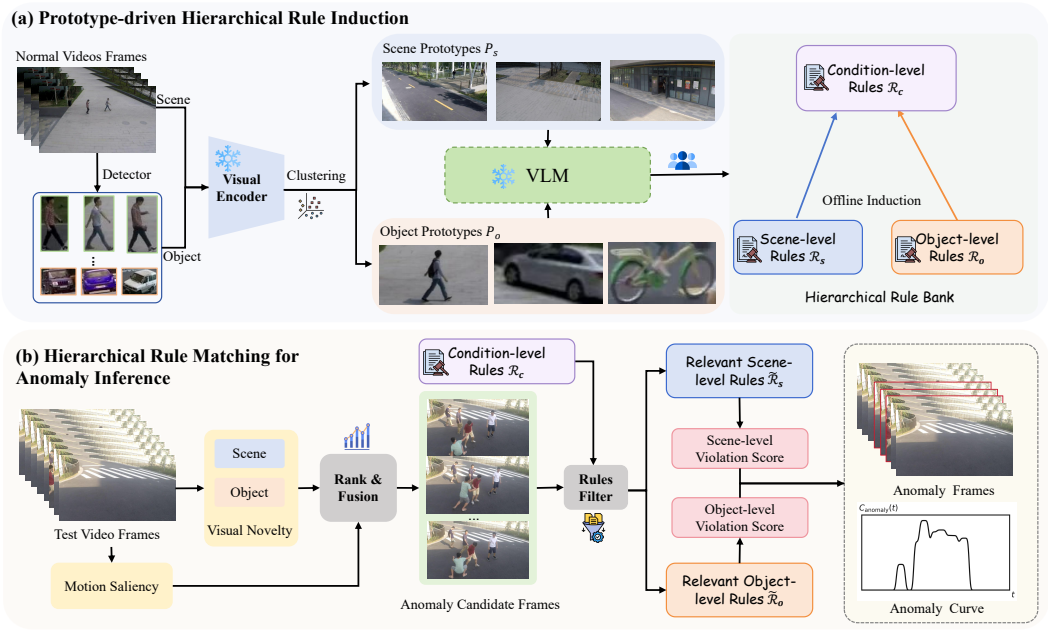


Figure 1: Overall framework of ProtoRule-VAD. The framework consists of two stages: prototype-driven hierarchical rule induction and selective hierarchical rule matching. In the offline stage, normal videos are summarized into scene and object prototypes, from which a pretrained VLM induces a hierarchical rule bank containing scene-level, object-level, and condition-level rules. During inference, motion saliency and visual novelty are fused to prioritize candidate segments. For each candidate, condition-level rules guide semantic routing to context-compatible scene-level and object-level rules, whose rule violation scores are aggregated to produce the final anomaly score.

errors (Hasan et al., 2016; Gong et al., 2019a; Park et al., 2020; Liu et al., 2021; Cai et al., 2021; Liu et al., 2018), or deviations from learned normal distributions (Ionescu et al., 2019; Georgescu et al., 2021; Ramachandra et al., 2020; Wang et al., 2020). Weakly supervised methods formulate VAD as video-level classification or multiple instance learning (Sultani et al., 2018b; Lv et al., 2023; Zhang et al., 2023), and learn segment-level anomaly scores through temporal aggregation, feature magnitude modeling, or discriminative representation learning (Tian et al., 2021; Chen et al., 2023; Feng et al., 2021; Liu et al., 2023b; Yang et al., 2024b; Pu et al., 2024). Despite their effectiveness, these visual feature-space methods provide limited semantic interpretability and often struggle with cross-scene and open-category anomalies.

To address these limitations, Recent advances in CLIP (Wu et al., 2024; Zanella et al., 2024a), vision-language models, and large multimodal models (Liu et al., 2023a; Lin et al., 2024) have enabled open-vocabulary and interpretable VAD. Several works adapt pretrained multimodal models for anomaly understanding through fine-tuning or semantic reasoning. VADor (Lv & Sun, 2024) and Holmes-VAU (Zhang et al., 2025) explore video-language reasoning for anomaly detection, while other methods leverage frozen foundation models through prompting and inference-time reasoning. LAVAD (Zanella et al., 2024b) performs anomaly analysis based on generated visual descriptions, and Event-VAD (Shao et al., 2025) introduces event-level modeling to improve temporal reasoning. Although these approaches enhance semantic interpretability, many of them still require dense semantic generation or extensive reasoning over video frames, resulting in considerable inference costs.

Training-free VAD with context-aware anomaly reasoning. Recent studies exploit pretrained multimodal models to perform VAD without anomalous supervision or model fine-tuning. AnomalyRuler (Yang et al., 2024a) induces normality rules from a small number of normal reference samples and performs rule-based anomaly discrimination, demonstrating the potential of explicit semantic knowledge for VAD. Flashback (Lee et al., 2025) further separates semantic knowledge construction from online inference by building an offline memory and performing retrieval-based anomaly detection without

repeated LLM invocation. Other methods explore flexible semantic reasoning, such as customizable anomaly concepts in AnyAnomaly (Ahn et al., 2026), question-guided reasoning in VERA (Ye et al., 2025), and temporal hierarchy modeling in VADTree (Li et al., 2025). Different from these methods, ProtoRule-VAD focuses on constructing a compact and hierarchical semantic representation from normal videos. Specifically, we first compress normal videos into visual prototypes and then induce scene-level, object-level, and condition-level rules. The condition-level rules further provide context-aware routing for selective hierarchical rule matching, enabling efficient anomaly inference without dense online semantic reasoning.

3 METHOD

3.1 PROTOTYPE-DRIVEN HIERARCHICAL RULE INDUCTION

Normal Prototype Construction. Let the normal training set be $\mathcal{D}_n = \{V_i^n\}_{i=1}^N$. For a sampled video clip x_t, we extract its visual feature using a frozen visual encoder $E(x_t)$:

$$f_t = E(x_t), \quad f_t \in \mathbb{R}^d \tag{1}$$

Directly feeding all normal video into a VLM would repeatedly process highly similar visual states. We therefore summarize the normal data in visual feature space before semantic rule induction. Specifically, K-Means clustering is applied to the global visual representations to construct a scene prototype bank:$P_s = \{p_k^s\}_{k=1}^{K_s}$.Each prototype represents a recurring global scene configuration in the normal videos. For every cluster, the clip whose feature is closest to the cluster centroid is retained as its representative video clips.

Global scene features alone, however, may suppress foreground variations that are important for video anomaly detection. We therefore construct an object prototype bank at the object level which derived from object-region features. The object regions are extracted by detector and encoded by the same visual encoder. For each object category (c), the corresponding object features are clustered to form an object prototype bank:$P_o = \{p_m^o\}_{m=1}^{K_o}$.For each prototype, the feature nearest to the respective cluster centroids is retained as representative video clips.

The prototypes are not used as learned anomaly classifiers. Instead, they provide a compact visual summary of recurring normal states and form the interface between dense normal video and subsequent semantic rule induction. Consequently, only a small collection of representative segments needs to be processed by the VLM.

Contrastive Hierarchical Rule Induction. Prototype construction summarizes what is visually representative in the normal data, but visual prototypes alone do not explicitly describe what behavior is expected in a scene or what constitutes a context-dependent violation. We therefore convert the representative prototype frames into structured textual knowledge using a frozen pretrained VLM.

Let $\mathcal{I}_p$ denote the representative video frames associated with the prototypes, and let $\mathcal{T}$ denote the contrastive prompt. Rule induction is written as

$$\mathcal{R} = M_{\text{vlm}}(\mathcal{I}_p, \mathcal{T}) \tag{2}$$

Rather than asking the VLM for an unrestricted description of each representative frames, we organize the generated knowledge along two complementary axes. First, we explicitly separate different semantic granularities. The resulting hierarchical rule bank is $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$,where $\mathcal{R}_s$, $\mathcal{R}_o$, and $\mathcal{R}_c$ denote scene-level, object-level, and condition-level rules, respectively.

- **Scene-level rules** $\mathcal{R}_s$: The VLM describes the global scene context together with behaviors typically observed under normal conditions. It is additionally asked to hypothesize behaviors or events that would constitute potential violations under the same context.
- **Object-level rules** $\mathcal{R}_o$: Representative object prototypes are used to elicit descriptions of its normal appearances, states, and behaviors, together with possible abnormal states or actions.Object-level rules complement the global scene rules by preserving localized behavioral information that may be diluted in whole-scene representations.
- **Condition-level rules** $\mathcal{R}_c$:An anomaly is often defined not by the presence of an object alone but by the compatibility between the object and its surrounding scene. We therefore

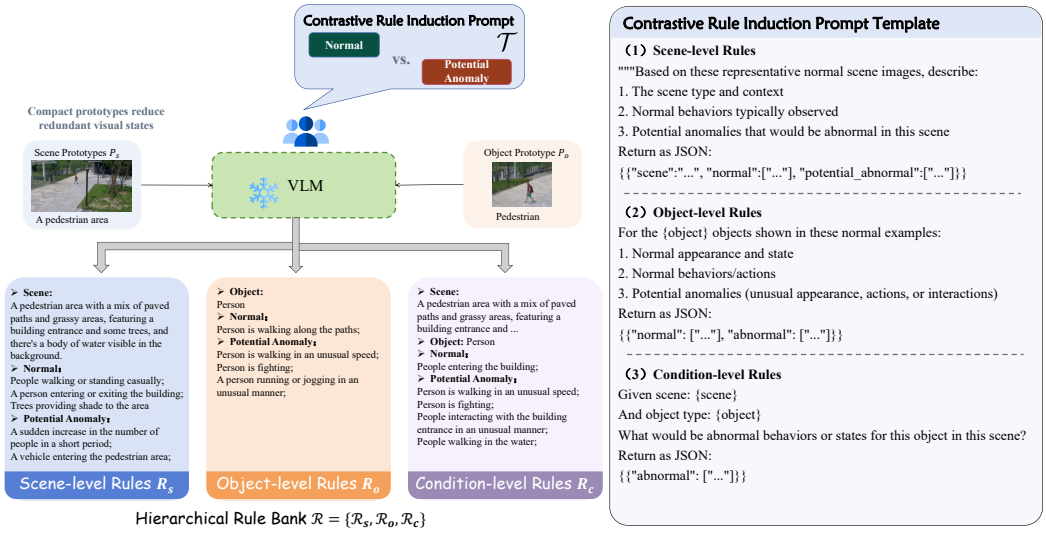


Figure 2: Prototype-driven contrastive hierarchical rule induction. Representative frames associated with scene-level and object-level normal prototypes are fed into a pretrained VLM with structured contrastive prompts to construct a hierarchical rule bank $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$.

additionally induce rules conditioned jointly on a scene context and an object category. These rules encode scene-object compatibility and potential context-dependent violations.

Second, our prompt asks the VLM to describe both normal patterns and potential violations. We refer to this organization as contrastive rule induction. Unlike anomaly supervision, these potential violation rules are not derived from anomalous training video and are not treated as ground-truth anomaly annotations. The resulting rule bank therefore transforms a compact collection of normal video into structured semantic knowledge.

3.2 Hierarchical Rule Matching with Segment Prescreening for Anomaly Inference

To avoid exhaustive semantic matching on every video segment during inference, ProtoRule-VAD separates where to perform semantic verification from how to determine semantic violations.The first step uses lightweight visual cues to prioritize candidate segments. The second step performs context-aware rule matching only on these selected candidates. Motion saliency and prototype novelty determine which portions of the video receive further semantic examination, while the anomaly score itself is produced by the rule-matching stage.

Candidate Segment Prescreening with Motion and Prototypes. Given a test video, we divide it into overlapping temporal segments. For each segments, two complementary visual cues are computed: motion saliency and prototype-based visual novelty.

The motion saliency s_t^m measures the fine-grained temporal dynamics within the segment. Specifically, we traverse all consecutive frame pairs in the segment and calculate the pixel-wise absolute difference for each pair. The final motion score is defined as the average of these differences across all pairs:

$$s_t^m = \frac{1}{(T-1)HW} \sum_{\tau=2}^{T} \sum_{u,v} |x_\tau(u,v) - x_{\tau-1}(u,v)| \tag{3}$$

where T is the total number of frames within the segment, and x_τ denotes the τ-th frame. However, the motion saliency does not characterize whether the visual feature is compatible with the normal states observed during offline adaptation. We therefore compute the visual novelty using the normal prototype banks.

For the global scene features f_t of segment, the scene-level novelty score is defined by its Euclidean distance to the closest normal scene prototype; for the detected foreground objects, we measure object-level deviation from the corresponding normal object prototypes:

$$s_t^n = \min_k \|f_t - p_k^s\|_2, \qquad s_t^o = \frac{1}{J_t}\sum_{j=1}^{J_t} \min_m \|f_{t,j}^o - p_m^o\|_2. \tag{4}$$

$f_{t,j}^o$ denote the local object features of the segment, respectively. p_k^s and p_m^o refer to the scene and object prototypes, and J_t is the number of detected objects.The two prototype-based deviations are combined.Both s_t^m and $\tilde{s}_t^n$ indicate the degree of deviation from normalcy. To robustly fuse these heterogeneous signals, we sort all segments in descending order based on their anomaly degrees, assigning rank 1 to the most anomalous segment for each signal.

$$\tilde{s}_t^n = s_t^n + s_t^o, \qquad r_t = \min\left(\mathrm{Rank}(s_t^m), \mathrm{Rank}(\tilde{s}_t^n)\right). \tag{5}$$

We then compute the final fused rank r_t.The minimum operator implements an OR-style prioritization: a segment receives a favorable fused rank when it is salient under either temporal change or prototype deviation. Finally, segments with the top-M smallest r_t values are selected to form the candidate set $\mathcal{C}$.

Condition-Guided Semantic Routing. A straightforward rule-based inference scheme could compare every candidate against the entire rule bank. However, such exhaustive comparison mixes rules generated for different scene-object configurations. ProtoRule-VAD therefore adopts a coarse-to-fine hierarchical matching strategy that perform condition-guided rule routing before calculating rule violation scores.

For each candidate segment $t \in \mathcal{C}$, we obtain its global scene context and detect the foreground object categories.The condition-level rule bank $\mathcal{R}_c$ is then used to identify the scene-object conditions associated with the current feature and to filter the corresponding subsets of scene-level rules $\widetilde{\mathcal{R}}_s$ and object-level rules $\widetilde{\mathcal{R}}_o$:

$$\widetilde{\mathcal{R}}_{s,t} \subseteq \mathcal{R}_s, \qquad \widetilde{\mathcal{R}}_{o,t,j} \subseteq \mathcal{R}_o. \tag{6}$$

In this way, $\mathcal{R}_c$ functions as a semantic router rather than as an additional scoring branch. The routing step restricts subsequent comparisons to rules associated with the current scene context and detected object categories, thereby reducing irrelevant cross-context rule comparisons.

Rule Matching And Anomaly Scoring. The prescreening scores characterize visual changes and deviations from representative normal prototypes, but such visual deviations do not directly specify whether the observed content violates the semantic regularities of the current context. We therefore use the hierarchical rule bank to perform context-aware semantic discrimination on the selected candidate segments.After condition-guided routing, each candidate is compared only with scene-level and object-level rules associated with its current context. Rather than directly treating similarity to a potential-violation rule as the anomaly score, we formulate rule matching as a relative comparison between two semantic hypotheses: consistency with the normal rules and consistency with the potential-violation rules. This design follows our contrastive rule induction, where normal patterns and potential violations are jointly constructed for the same semantic context.

$$s_{t,\text{scene}}^r = s_{t,\text{scene}}^{r,a} - s_{t,\text{scene}}^{r,n}. \tag{7}$$

Here, $s_{t,\text{scene}}^{r,a}$ measures the compatibility between the candidate and the routed potential violation rules, while $s_{t,\text{scene}}^{r,n}$ measures its compatibility with the corresponding normal rules.A larger positive margin indicates that the segment is semantically more similar with the potential violation rules than with the corresponding normal rules.Compared with using the absolute similarity to a potential violation rule alone, this relative formulation explicitly takes the normal rules into account.

The same relative comparison is performed at the object-level. Scene-level and object-level matching provide complementary semantic evidence: the former captures global contextual consistency, whereas the latter preserves localized object states and behaviors that may be less evident in the global representation.The object-level violation score is as follow:

$$s_{t,j,\text{obj}}^r = s_{t,j,\text{obj}}^{r,a} - s_{t,j,\text{obj}}^{r,n}. \tag{8}$$

A segment-level anomaly does not require simultaneous violations at all semantic granularities. A strong contextual violation at the scene level or a localized violation associated with a single foreground object can both provide relevant anomaly evidence. We therefore aggregate the two granularities using the maximum score,the final candidate score s_t^r is defined as:

$$s_t^r = \max\left(s_{t,\text{scene}}^r, \max_j s_{t,j,\text{obj}}^r\right). \tag{9}$$

The max operator preserves the strongest available rule violation evidence and avoids introducing an additional learned fusion module, which is consistent with our training-free inference setting. Finally, the candidate gate and semantic rule matching are combined into the segment-level anomaly output: Thus, non-candidate segments do not invoke semantic rule matching, while candidate segments receive positive anomaly scores only when their potential violation similarity exceeds their corresponding normal rule similarity.Details can be found in Algorithm 2 of Appendix A.

$$S_t = \begin{cases} s_t^r, & t \in \mathcal{C} \text{ and } s_t^r > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

4 Experiments

4.1 Experimental Setup

We evaluate our method on **ShanghaiTech** (Luo et al., 2017), **UCF-Crime** (Sultani et al., 2018a), and **XD-Violence**. Following standard evaluation protocols, we employ frame-level Area Under the Receiver Operating Characteristic Curve (AUC) as the primary evaluation metric.In the offline stage, we use CLIP (ViT-B/32) and YOLOv8n to extract visual features and construct the prototype bank via K-Means clustering.We adopt Qwen2.5-VL-7B (Bai et al., 2025) as the vision-language model for offline rule induction.The maximum number of scene prototypes is set to $K_s = 8$, while the maximum number of object prototypes per category is set to $K_o = 8$ for ShanghaiTech and $K_o = 10$ for UCF-Crime and XD-Violence. All experiments are conducted on a single NVIDIA RTX A6000 GPU.Detailed dataset statistics and implementation details are provided in Appendix A.3.

4.2 Comparison with State-of-the-Art Methods

Table 1 compares ProtoRule-VAD with representative video anomaly detection methods under different supervision and optimization settings. Among training-free multimodal methods, ProtoRule-VAD achieves the best performance on UCF-Crime and ShanghaiTech among the compared approaches. On XD-Violence, it reaches 88.72% AUC, outperforming VERA, Event-VAD, LAVAD, and DR-VAD, while remaining below VAD-Tree.Compared with methods that require task-specific parameter optimization, ProtoRule-VAD also shows competitive performance despite using neither anomalous supervision nor model fine-tuning. For example, on UCF-Crime, ProtoRule-VAD achieves 85.68% AUC, which is close to VADor (85.90%) while requiring no parameter updates. These results indicate that converting normal observations into reusable hierarchical semantic rules can provide strong anomaly discrimination under a training-free setting.Unlike approaches that repeatedly invoke VLMs or MLLMs during inference, ProtoRule-VAD performs expensive semantic knowledge construction offline and reuses the induced hierarchical rule bank during online inference. This separation enables the method to retain semantic interpretability while substantially reducing test-time computation.

4.3 Ablation Studies

Effectiveness of Core Module. Table 2a evaluates the major components of ProtoRule-VAD. The baseline, which performs frame-wise VLM description and rule generation, achieves an AUC of 79.31%. Introducing normal prototype construction improves the performance to 82.72%, corresponding to an AUC gain of +3.41%.. This result supports the use of representative prototypes as a compact input for subsequent semantic rule induction. Adding contrastive rule induction further improves the AUC from 82.72% to 84.33%. By organizing the rule bank into normal patterns and potential violations, the induced textual knowledge provides more discriminative references for subsequent visual-text comparison. Finally, incorporating selective hierarchical rule matching increases the

Table 1: AUC (%) on benchmark datasets UCF-Crime (UCF-C), ShanghaiTech Campus (ShT), and XD-Violence (XD-Vio). * indicates our reproduced results under the same experimental setting. Other results are cited from their papers.

Mode	Methods	Explainable	UCF-C	ShT	XD-Vio
Self-Supervised / One-Class	GODS (Wang & Cherian, 2019)	×	70.46	–	61.56
	S3R (Wu et al., 2022)	×	77.15	79.89	–
	FPDM (Yan et al., 2023)	×	74.70	78.60	–
	MULDE (Micorek et al., 2024)	×	78.50	81.30	–
Unsupervised	GCL (Zaheer et al., 2022)	×	71.04	78.93	–
	DYANNET (Thakare et al., 2023)	×	84.50	–	–
Fine-Tuning Multimodal VAD	VADor (Lv & Sun, 2024)	✓	85.90	–	–
	Holmes-VAU (Zhang et al., 2025)	✓	88.96	–	–
	LAVIDA (Dai et al., 2026)	✓	82.18	85.28	–
Training-free Multimodal VAD	ZS CLIP (Radford et al., 2021)	✓	53.16	–	38.21
	ZS IMAGEBIND-I (Girdhar et al., 2023)	✓	53.65	–	58.81
	ZS IMAGEBIND-V (Girdhar et al., 2023)	✓	55.78	–	55.06
	LLAVA-1.5 (Liu et al., 2024)	✓	72.84	–	79.62
	AnomalyRuler (Yang et al., 2024a)	✓	–	85.20	–
	LAVAD (Zanella et al., 2024b)	✓	80.28	–	85.36
	VERA* (Ye et al., 2025)	✓	83.50	–	87.05
	Event-VAD (Shao et al., 2025)	✓	82.03	–	87.51
	VAD-Tree (Li et al., 2025)	✓	84.74	–	**90.44**
	PANDA (Yang et al., 2025)	✓	84.89	–	–
	DR-VAD (Pei et al., 2026)	✓	79.44	–	87.45
	AnyAnomaly (Ahn et al., 2026)	✓	80.70	79.70	–
	ProtoRule-VAD	✓	**85.68**	**86.53**	88.72

performance to 86.53%, yielding a further improvement of 2.20 percentage points. The progressive gains indicate that prototype compression, contrastive rule construction, and context-aware rule matching provide complementary contributions to the overall framework.

Effectiveness of Candidate Prioritization Signals. Table 2b studies the visual cues used before semantic rule matching. Motion saliency alone achieves 80.57% AUC, while prototype-based visual novelty alone obtains 78.38%. Combining the two visual cues increases the AUC to 83.94%, showing that they capture complementary properties of suspicious segments: motion saliency emphasizes temporal changes, whereas prototype novelty measures deviations from representative normal visual states. When rule-violation scoring is further applied to the prioritized candidates, the performance increases from 83.94% to 86.53%. This improvement shows that the visual cues and semantic rule matching play different roles in the inference pipeline. The former determines which segments receive further examination, whereas the latter provides context-dependent semantic evidence for anomaly scoring.

Table 2: Ablation studies of ProtoRule-VAD on ShanghaiTech using frame-level AUC (%): (a) core component ablation; (b) candidate prioritization and rule-violation scoring ablation; and (c) hierarchical rule usage ablation.

Prototype	Contrastive Rule	Hierarchical Match	AUC
×	×	×	79.31
✓	×	×	82.72
✓	✓	×	84.33
✓	✓	✓	**86.53**

(a)

Motion	Novelty	Rule Violation	AUC
✓	×	×	80.57
×	✓	×	78.38
✓	✓	×	83.94
✓	✓	✓	**86.53**

(b)

Scene -level	Object -level	Condition -level	AUC
✓	×	×	78.63
×	✓	×	83.57
✓	✓	×	84.83
✓	✓	✓	**86.53**

(c)

Inference-stage Rules Usage Ablation. Table 2c validates the necessity of our hierarchical rule design. Relying solely on scene ($\mathcal{R}_s$) or object ($\mathcal{R}_o$) rules yields sub-optimal AUCs of 78.63%

and 83.57%, as they respectively miss fine-grained foreground anomalies and ignore environmental context. Naively combining both improves the AUC to 84.83% but inevitably introduces context-mismatch errors by blindly matching all available rules without constraints. Introducing condition-level rules ($\mathcal{R}_c$) yields the best performance (86.53%). By serving as a semantic router, $\mathcal{R}_c$ dynamically filters the rule bank to select only the scene and object rules strictly relevant to the current context. This initial screening explicitly models scene-object interactions and resolves the context-mismatch problem, significantly reducing false alarms in complex scenarios.We further analyze the representativeness of prototype-based selection against random sampling in Appendix C.1, including feature-space coverage and qualitative visualization.

4.4 Efficiency Analysis

Figure 3 compares the detection performance and online inference throughput of representative VLM-based VAD methods on UCF-Crime. For a fair comparison, all methods are evaluated under the same hardware setting using a single NVIDIA A6000 GPU, and FPS is measured during online inference. ProtoRule-VAD achieves an AUC of 85.68% while processing videos at 562 FPS, substantially outperforming all compared methods in inference throughput. In particular, the fastest competing method, LAVAD, reaches 72 FPS, whereas other VLM/MLLM-based approaches operate at only 2–8 FPS under the same setting. ProtoRule-VAD therefore provides approximately $7.8\times$ higher throughput than the fastest baseline in Figure 3, while maintaining competitive anomaly detection performance. The efficiency gain mainly stems from the separation between offline semantic knowledge construction and selective online rule matching. The pretrained VLM is invoked during the offline stage to induce reusable hierarchical rules from compact normal prototypes, rather than being repeatedly queried throughout test-time inference. During online inference, lightweight motion saliency and prototype-based visual novelty first prioritize candidate segments, after which semantic rule matching is selectively performed on these candidates. This design substantially reduces the amount of expensive semantic processing required during inference. These results show that ProtoRule-VAD provides a better balance between detection accuracy and practical inference efficiency.

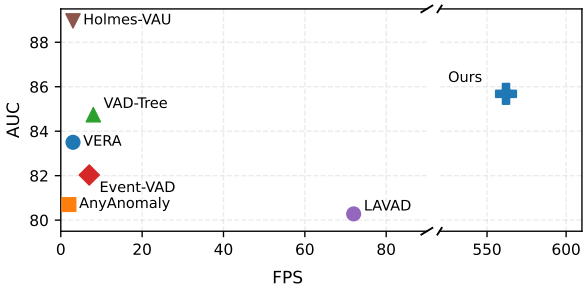


Figure 3: Efficiency comparison on UCF-Crime under the single-A6000 setting.

5 Conclusion

We presented ProtoRule-VAD, a video anomaly detection framework that converts normal videos into reusable hierarchical semantic rules without anomalous supervision or model parameter updates. In the offline stage, representative scene-level and object-level prototypes are constructed from normal videos and used to induce hierarchical rules with a pretrained VLM. During inference, motion saliency and visual novelty first prioritize candidate segments, after which selective hierarchical rule matching is performed. Condition-level rules serve as semantic routers to retrieve context-compatible scene-level and object-level rules, whose violation scores are used for anomaly prediction.Experiments on UCF-Crime, ShanghaiTech, and XD-Violence show that ProtoRule-VAD achieves competitive detection performance while substantially improving online inference throughput. ProtoRule-VAD suggests that separating offline semantic knowledge construction from selective online rule matching provides an effective way to incorporate structured semantic knowledge into efficient video anomaly detection.

AI USE STATEMENT

Generative AI tools were used to assist with code debugging and translation. They were additionally used for literature search and language polishing of the manuscript. All AI-assisted outputs were manually reviewed and verified. We take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses existing video anomaly detection benchmarks and does not involve the collection of new human-subject data. The proposed method is intended for research purposes. As video anomaly detection may be applied to surveillance scenarios, practical deployment should carefully consider privacy, fairness, and potential misuse.

REPRODUCIBILITY STATEMENT

We provide the main implementation details and experimental settings in Section 4.1. Detailed algorithms for both offline hierarchical rule induction and online anomaly inference are provided in Appendix A. Appendix B reports the exact VLM prompts, generation configuration, call protocol, representative structured outputs, and rule parsing procedure used to construct the hierarchical rule bank. Additional analyses of the induced rules and inference behavior are provided in Appendices C and D. These materials are included to facilitate reproduction and verification of the proposed method and experimental results.

REFERENCES

Sunghyun Ahn, Youngwan Jo, Kijung Lee, Sein Kwon, Inpyo Hong, and Sanghyun Park. Anyanomaly: Zero-shot customizable video anomaly detection with lvlm. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3026–3035, 2026.

Ghazal Alinezhad Noghre, Armin Danesh Pazho, and Hamed Tabkhi. A survey on video anomaly detection via deep learning: Human, vehicle, and environment. *arXiv preprint arXiv:2508.14203*, 2025. doi: 10.48550/arXiv.2508.14203.

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. doi: 10.48550/arXiv.2502.13923.

Ruichu Cai, Hao Zhang, Wen Liu, Shenghua Gao, and Zhifeng Hao. Appearance-motion memory consistency network for video anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 938–946, 2021.

Yingxian Chen, Zhengzhe Liu, Baoheng Zhang, Wilton Fok, Xiaojuan Qi, and Yik-Chung Wu. Mgfn: Magnitude-contrastive glance-and-focus network for weakly-supervised video anomaly detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 387–395, 2023.

Zunkai Dai, Ke Li, Jiajia Liu, Jie Yang, and Yuanyuan Qiao. No need for real anomaly: Mllm empowered zero-shot video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35648–35658, June 2026.

Jia-Chang Feng, Fa-Ting Hong, and Wei-Shi Zheng. Mist: Multiple instance self-training framework for video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14009–14018, 2021.

Mariana-Iuliana Georgescu, Antonio Barbalau, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. Anomaly detection in video via self-supervised and multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12742–12752, June 2021.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15180–15190, 2023.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1705–1714, October 2019a.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1705–1714, 2019b.

Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 733–742, 2016.

Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7842–7851, June 2019.

Hyogun Lee, Haksub Kim, Ig-Jae Kim, and Yonghun Choi. Flashback: Memory-driven zero-shot, real-time video anomaly detection. *arXiv preprint arXiv:2505.15205*, 2025.

Wenlong Li, Yifei Xu, Yuan Rao, Zhenhua Wang, and Shuiguang Deng. Vadtree: Explainable training-free video anomaly detection via hierarchical granularity-aware tree. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

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. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5971–5984, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.342. URL https://aclanthology.org/2024.emnlp-main.342/.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024.

Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6536–6545, 2018.

Yang Liu, Siao Liu, Xiaoguang Zhu, Jielin Li, Hao Yang, Liangyu Teng, Junchen Guo, Yan Wang, Dingkang Yang, and Jing Liu. Privacy-preserving video anomaly detection: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Zhian Liu, Yongwei Nie, Chengjiang Long, Qing Zhang, and Guiqing Li. A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13588–13597, October 2021.

Zuhao Liu, Xiao-Ming Wu, Dian Zheng, Kun-Yu Lin, and Wei-Shi Zheng. Generating anomalies for video anomaly detection with prompt-based feature mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24500–24510, June 2023b.

Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *Proceedings of the IEEE international conference on computer vision*, pp. 341–349, 2017.

Hui Lv and Qianru Sun. Video anomaly detection and explanation via large language models. *arXiv preprint arXiv:2401.05702*, 2024.

Hui Lv, Zhongqi Yue, Qianru Sun, Bin Luo, Zhen Cui, and Hanwang Zhang. Unbiased multiple instance learning for weakly supervised video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8022–8031, June 2023.

Jakub Micorek, Horst Possegger, Dominik Narnhofer, Horst Bischof, and Mateusz Kozinski. Mulde: Multiscale log-density estimation via denoising score matching for video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18868–18877, 2024.

Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14372–14381, June 2020.

Yanting Pei, Minhao Hao, Yaping Huang, and Qi Zou. Dr-vad: Definition-guided reasoning for training-free video anomaly detection. *Neurocomputing*, pp. 132836, 2026.

Yujiang Pu, Xiaoyu Wu, Lulu Yang, and Shengjin Wang. Learning prompt-enhanced context features for weakly-supervised video anomaly detection. *IEEE Transactions on Image Processing*, 33: 4923–4936, 2024. doi: 10.1109/TIP.2024.3451935.

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 *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Bharathkumar Ramachandra, Michael J. Jones, and Ranga Raju Vatsavai. Learning a distance function with a siamese network to localize anomalies in videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2598–2607, 2020.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4393–4402. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/ruff18a.html`.

Yihua Shao, Haojin He, Sijie Li, Siyu Chen, Xinwei Long, Fanhu Zeng, Yuxuan Fan, Muyang Zhang, Ziyang Yan, Ao Ma, et al. Eventvad: Training-free event-aware video anomaly detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2586–2595, 2025.

Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6479–6488, 2018a.

Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6479–6488, 2018b.

Che Sun, Yunde Jia, Yao Hu, and Yuwei Wu. Scene-aware context reasoning for unsupervised abnormal event detection in videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 184–192, 2020. doi: 10.1145/3394171.3413887.

Jiaqi Tang, Hao Lu, Ruizheng Wu, Xiaogang Xu, Ke Ma, Cheng Fang, Bin Guo, Jiangbo Lu, Qifeng Chen, and Ying-Cong Chen. Hawk: Learning to understand open-world video anomalies. *Advances in Neural Information Processing Systems*, 37:139751–139785, 2024.

Kamalakar Vijay Thakare, Yash Raghuwanshi, Debi Prosad Dogra, Heeseung Choi, and Ig-Jae Kim. Dyannet: A scene dynamicity guided self-trained video anomaly detection network. In *Proceedings of the IEEE/CVF Winter conference on applications of computer vision*, pp. 5541–5550, 2023.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4975–4986, 2021.

Jue Wang and Anoop Cherian. Gods: Generalized one-class discriminative subspaces for anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8201–8211, 2019.

Ziming Wang, Yuexian Zou, and Zeming Zhang. Cluster attention contrast for video anomaly detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2463–2471, 2020. doi: 10.1145/3394171.3413529.

Jhih-Ciang Wu, He-Yen Hsieh, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Self-supervised sparse representation for video anomaly detection. In *European Conference on Computer Vision*, pp. 729–745. Springer, 2022.

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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 6074–6082, 2024.

Cheng Yan, Shiyu Zhang, Yang Liu, Guansong Pang, and Wenjun Wang. Feature prediction diffusion model for video anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5527–5537, 2023.

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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024a.

Zhiwei Yang, Jing Liu, and Peng Wu. Text prompt with normality guidance for weakly supervised video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18899–18908, June 2024b.

Zhiwei Yang, Chen Gao, and Mike Zheng Shou. Panda: Towards generalist video anomaly detection via agentic ai engineer. In *NeurIPS*, 2025.

Muchao Ye, Weiyang Liu, and Pan He. Vera: Explainable video anomaly detection via verbalized learning of vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8679–8688, 2025.

M Zaigham Zaheer, Arif Mahmood, M Haris Khan, Mattia Segu, Fisher Yu, and Seung-Ik Lee. Generative cooperative learning for unsupervised video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14744–14754, 2022.

Luca Zanella, Benedetta Liberatori, Willi Menapace, Fabio Poiesi, Yiming Wang, and Elisa Ricci. Delving into clip latent space for video anomaly recognition. *Computer Vision and Image Understanding*, 249:104163, 2024a. doi: 10.1016/j.cviu.2024.104163.

Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. Harnessing large language models for training-free video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18527–18536, 2024b.

Chen Zhang, Guorong Li, Yuankai Qi, Shuhui Wang, Laiyun Qing, Qingming Huang, and Ming-Hsuan Yang. Exploiting completeness and uncertainty of pseudo labels for weakly supervised video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16271–16280, June 2023.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

A Detailed Method and Algorithms

A.1 Offline Prototype-Driven Hierarchical Rule Induction

Algorithm 1 summarizes the offline knowledge construction stage of ProtoRule-VAD. Given normal training videos, we first extract global scene representations and category-specific object representations using a frozen visual encoder and an object detector. K-Means clustering is then performed in the corresponding feature spaces to construct the scene prototype bank $\mathcal{P}_s$ and object prototype bank $\mathcal{P}_o$. For each cluster, the real normal sample closest to the cluster centroid is retained as its representative example, allowing the normal training data to be summarized by a compact set of recurring visual states.

The representative scene and object prototypes are subsequently processed by a frozen pretrained VLM with structured contrastive prompts. Scene-level rules $\mathcal{R}_s$ describe the global scene context together with normal patterns and potential violations, while object-level rules $\mathcal{R}_o$ characterize normal and potentially abnormal states or behaviors of foreground objects. In addition, condition-level rules $\mathcal{R}_c$ are induced from scene–object combinations to encode context-dependent compatibility. The resulting hierarchical rule bank $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$, together with the prototype banks, is constructed once offline and reused during test-time inference.

Algorithm 1: Offline Prototype-Driven Hierarchical Rule Induction

Input: Normal training videos $\mathcal{D}_n$, frozen visual encoder $\mathcal{E}$, object detector $\mathcal{D}$, pretrained VLM $\mathcal{M}_{\text{vlm}}$, prototype numbers K_s and K_o
Output: Scene and object prototype banks $\mathcal{P}_s, \mathcal{P}_o$, hierarchical rule bank $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$
Sample normal video clips from $\mathcal{D}_n$;
for *each sampled normal clip* x_t **do**
  Extract global scene feature $f_t = \mathcal{E}(x_t)$;
  Detect foreground objects and extract category-specific object features $\{(l_{t,j}, f^o_{t,j})\}$;
end
Apply K-Means to global scene features and construct $\mathcal{P}_s = \{p^s_k\}_{k=1}^{K_s}$;
For each scene cluster, retain the real normal sample nearest to its centroid as the representative scene example;
for *each object category* c **do**
  Cluster its object features and construct category-specific object prototypes;
  Retain the real object samples nearest to the corresponding centroids as representative object examples;
end
Collect all object prototypes to form $\mathcal{P}_o$;
Initialize $\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c \leftarrow \emptyset$;
for *each representative scene prototype* **do**
  Apply the scene-level contrastive prompt to $\mathcal{M}_{\text{vlm}}$;
  Generate scene context, normal-pattern, and potential-violation rules and add them to $\mathcal{R}_s$;
end
for *each object category* **do**
  Apply the object-level contrastive prompt to its representative object prototypes;
  Generate normal and potential-violation rules and add them to $\mathcal{R}_o$;
end
for *each scene–object condition represented by the normal prototypes* **do**
  Apply the condition-level prompt using the corresponding scene context and object category;
  Generate context-dependent rules and add them to $\mathcal{R}_c$;
end
Construct $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$;
return $\mathcal{P}_s$, $\mathcal{P}_o$, *and* $\mathcal{R}$;

A.2 Hierarchical Rule Matching for Anomaly Inference

Algorithm 2 presents the complete online inference procedure. Given a test video, we first divide it into overlapping temporal segments and compute two lightweight visual cues for each segment:

motion saliency and prototype-based visual novelty. The novelty signal measures the deviation of the current scene and foreground objects from the normal prototype banks constructed during the offline stage. The rankings produced by the two cues are fused to prioritize suspicious segments, and only the top-ranked segments are retained as the candidate set $\mathcal{C}$.

For each candidate segment, condition-level rules $\mathcal{R}_c$ act as a semantic routing layer to identify the scene–object context and retrieve only the relevant subsets of scene-level and object-level rules. The candidate is then compared with the routed normal and potential-violation rules in the shared vision-language feature space. We compute the scene-level and object-level rule violation margins as the difference between their potential-violation and normal-rule similarities, and aggregate them using the maximum response. A candidate receives a positive anomaly score only when the resulting violation margin is positive; non-candidate segments bypass semantic rule matching and are assigned zero anomaly scores. In this way, expensive semantic verification is restricted to a small set of visually suspicious segments while retaining context-aware hierarchical reasoning.

Algorithm 2: Video Anomaly Detection Algorithm Based on Multi-Signal Fusion and Hierarchical Rule Matching

Input: Inference video V, scene and object prototype banks $\mathcal{P}_s, \mathcal{P}_o$, hierarchical rule bank $\mathcal{R} = \{\mathcal{R}_s, \mathcal{R}_o, \mathcal{R}_c\}$, sliding window size W
Output: Anomaly detection results and confidence scores of all test segments
Extract video frame sequence and construct the sliding window set with window length W;
for *Each sliding window* w_t **do**
  Calculate the motion saliency score s_t^m within the window;
  Extract visual features and compute the fused visual novelty score $\tilde{s}_t^n$ against the normal prototype bank;
  Integrate the ranking quantiles of motion and novelty cues to select the high-recall candidate anomalous segment set $\mathcal{C}$;
end
for *Each candidate segment* $t \in \mathcal{C}$ **do**
  Extract global scene visual feature f_t;
  Leverage the object detector to extract foreground object categories $\{l_{t,j}\}$ and visual features $\{f_{t,j}^o\}$;
  Semantic Routing: Utilize condition-level rules $\mathcal{R}_c$ to filter the rule dictionary based on the identified scene context and object categories $\{l_{t,j}\}$, yielding relevant scene rules $\tilde{\mathcal{R}}_s \subseteq \mathcal{R}_s$ and object rules $\tilde{\mathcal{R}}_o \subseteq \mathcal{R}_o$;
  Compute scene violation margin $s_{t,\text{scene}}^r = s_{t,\text{scene}}^{r,a} - s_{t,\text{scene}}^{r,n}$ by measuring visual-text embedding similarities against $\tilde{\mathcal{R}}_s$;
  Initialize the maximum object violation score $s_{max}^{obj} = 0$;
  for *Each detected object* j **do**
    Retrieve category-specific rules for $l_{t,j}$ from the filtered subset $\tilde{\mathcal{R}}_o$;
    Compute object violation margin $s_{t,j,\text{obj}}^r = s_{t,j,\text{obj}}^{r,a} - s_{t,j,\text{obj}}^{r,n}$ using visual-text embedding similarities;
    $s_{max}^{obj} = \max(s_{max}^{obj}, s_{t,j,\text{obj}}^r)$;
  end
  Compute the comprehensive violation score of candidate segment: $s_t^r = \max(s_{t,\text{scene}}^r, s_{max}^{obj})$;
  if $s_t^r > 0$ **then**
    The segment is judged as anomalous, with confidence defined as s_t^r;
  else
    The segment is classified as normal;
  end
end
return Detection results of all candidate segments;

A.3 Implementation and Evaluation Details

Dataset statistics. We evaluate ProtoRule-VAD on three standard video anomaly detection benchmarks. **ShanghaiTech** (Luo et al., 2017) contains 437 videos, including 330 training videos and 107 testing videos collected from 13 diverse scenes. **UCF-Crime** (Sultani et al., 2018a) consists of 1,900 untrimmed real-world surveillance videos, with 1,610 videos for training and 290 videos for testing, covering 13 anomaly categories. **XD-Violence** contains 4,754 unedited videos, including 3,954 training videos and 800 testing videos, spanning approximately 217 hours and six categories of violent events. Following the standard protocols of these datasets, we report frame-level AUC as the evaluation metric.

Implementation details. For feature extraction, we use CLIP (ViT-B/32) and YOLOv8n to obtain scene-level and object-level visual representations from normal training videos. K-Means clustering is applied to construct scene and object prototype banks. The maximum number of scene prototypes is fixed to $K_s = 8$ for all datasets. The maximum number of object prototypes per category is set to $K_o = 8$ for ShanghaiTech and $K_o = 10$ for UCF-Crime and XD-Violence. We adopt Qwen2.5-VL-7B (Bai et al., 2025) as the frozen vision-language model for offline hierarchical rule induction. During inference, videos are processed using a sliding window of $W = 8$ frames with an overlap ratio of 0.5. All experiments are conducted on a single NVIDIA RTX A6000 GPU.The detailed VLM generation configuration, prompts, and structured outputs are provided in Appendix B.

B Details of Hierarchical Rule Induction

This section provides the exact prompts, call protocol, generation configuration, and representative structured outputs used to construct the hierarchical rule bank. The purpose is to make explicit which information is provided to the vision language model (VLM), which information is generated by the VLM, and how the generated text is converted into the scene-level, object-level, and condition-level rules used by ProtoRule-VAD. All rule induction calls use only representative samples selected from the normal training split. No anomalous training video, test video, frame-level anomaly annotation, or dataset anomaly label is inserted into the prompts.

B.1 Model Configuration and Common Message Format

We use Qwen2.5-VL-7B-Instruct as the frozen VLM and execute it locally with vLLM. The same system message and decoding configuration are used for all three types of rule-induction calls. Table 3 reports the complete generation configuration used in our implementation.The VLM is used only during offline rule induction; it is not invoked during online anomaly inference.

Table 3: VLM configuration used for offline hierarchical rule induction.

Item	Setting
Checkpoint	Qwen/Qwen2.5-VL-7B-Instruct
Inference engine	vLLM
Maximum generated tokens	2048
Repetition penalty	1.05
Maximum model length	8192 tokens
Maximum images per request	12
Output type	JSON object
Model update	None (frozen)

Every call is formatted as a two-message conversation. The common message structure is shown below, where `<IMAGE>` is repeated once for each visual input. Condition-level induction contains no image and therefore includes only the text item in the user message.

Listing 1: Common message structure for rule induction.

```
System:
```

```
You are a vision anomaly detector.

User:
<IMAGE_1> ... <IMAGE_N>
<LEVEL-SPECIFIC PROMPT>
```

The three calls use different visual inputs. For scene-level induction, each scene prototype is processed in an independent call containing its associated full-frame representative image. For object-level induction, all available representative crops of one detector category are grouped into one call. For condition-level induction, the generated scene description and an object category name are inserted into a text-only prompt. Thus, for K_s scene prototypes and C retained object categories, scene-level induction requires K_s visual calls, object-level induction requires at most C visual calls, and condition-level induction requires one text-only call for every retained scene–object pair.

B.2 Scene-Level Rule Induction

Input. For a scene prototype, the VLM receives one full-frame representative image selected as the real normal sample nearest to its K-Means centroid. The exact user prompt is:

Listing 2: Exact scene-level rule-induction prompt.

```
Based on these representative normal scene images, describe:
1. The scene type and context
2. Normal behaviors typically observed
3. Potential anomalies that would be abnormal in this scene

Return as JSON: {"scene": "...", "normal": ["..."],
"potential_abnormal": ["..."]}
```

Structured output. The following example corresponds to a normal ShanghaiTech scene prototype. For readability, only whitespace and line wrapping are normalized.

Listing 3: Representative parsed scene-level VLM output.

```
{
  "scene": "Public Square",
  "normal": [
    "People walking or standing casually",
    "Benches for sitting",
    "Trees providing shade",
    "Open space for leisure activities"
  ],
  "potential_abnormal": [
    "A sudden increase in the number of people",
    "Unusual behavior such as running or screaming",
    "Obstruction of pathways by objects or people",
    "Abandoned or damaged benches",
    "Significant changes in lighting or weather conditions"
  ]
}
```

The items under `normal` describe recurring observations in the normal prototype, whereas the items under `potential_abnormal` are hypotheses generated from the frozen model's prior knowledge. The latter are not obtained from anomalous videos and are not treated as ground-truth anomaly annotations.

B.3 Object-Level Rule Induction

Input. For each detector category, the VLM receives the representative object crops of that category. The placeholder `{object}` is replaced by the detector class name. The exact prompt is:

Listing 4: Exact object-level rule-induction prompt.

```
For the {object} objects shown in these normal examples:
1. Normal appearance and state
2. Normal behaviors/actions
3. Potential anomalies (unusual appearance, actions, or interactions)

Return as JSON: {"normal": ["..."], "abnormal": ["..."]}
```

For example, when inducing rules for the `person` category, the placeholder is replaced by `person`, and the request contains the normal person-prototype crops selected for that category.

Structured output. The following output is a whitespace-normalized reconstruction of the parsed fields stored for a representative ShanghaiTech `person` call:

Listing 5: Representative parsed object-level VLM output for the person category.

```
{
  "normal": [
    "The person in image 1 is walking with a backpack on their back.",
    "The person in image 6 is walking with a backpack on their back.",
    "The person in image 7 is walking with a backpack on their back.",
    "The person in image 8 is walking with a backpack on their back.",
    "The person in image 9 is walking with a backpack on their back.",
    "The person in image 10 is walking with a backpack on their back."
  ],
  "abnormal": [
    "The person in image 2 appears to be running, which is not typical for the
    context of walking with a backpack.",
    "The person in image 3 has an unusual posture, possibly indicating
    discomfort or an injury.",
    "The person in image 4 has an unusual posture, possibly indicating
    discomfort or an injury.",
    "The person in image 5 has an unusual posture, possibly indicating
    discomfort or an injury."
  ]
}
```

This example also illustrates why we retain both normal and potential-anomaly descriptions. A generated abnormal hypothesis is not directly accepted as a detection result. At inference time, its visual–text similarity must exceed the similarity to the corresponding normal description before it can produce a positive rule-violation margin.

B.4 Condition-Level Rule Induction

Input. Condition-level induction uses no visual input. Instead, a generated scene description and a detector category are inserted into the following exact text-only prompt:

Listing 6: Exact condition-level rule-induction prompt.

```
Given scene: {scene}
And object type: {object}

What would be abnormal behaviors or states for this object in this scene?
Return as JSON: {"abnormal": ["..."]}
```

For the running example, `{scene}` is replaced by `Public Square` and `{object}` is replaced by `person`. The complete instantiated user prompt is therefore:

Listing 7: Instantiated condition-level input for a Public Square–person pair.

```
Given scene: Public Square
And object type: person

What would be abnormal behaviors or states for this object in this scene?
Return as JSON: {"abnormal": ["..."]}
```

Structured output. The corresponding whitespace-normalized parsed output is:

Listing 8: Representative parsed condition-level VLM output.

```
{
  "abnormal": [
    "engaging in aggressive behavior towards others",
    "sudden and unprovoked outburst of anger",
    "disruptive actions that interfere with public order",
    "unattended or abandoned personal belongings in a public space",
    "sustained and inappropriate use of loud, disruptive speech"
  ]
}
```

Condition-level rules are indexed by the generated scene string and the detector category. During inference, this metadata associates a candidate segment with the scene- and object-specific subsets of the rule bank; the condition-level text itself is not added as an independent anomaly-scoring branch.

B.5 JSON Parsing and Conversion to Rule Entries

The decoder response is first parsed using a standard JSON parser. If the entire response is not valid JSON, the implementation extracts the substring between the first opening and final closing brace and attempts JSON parsing again.

For a valid response, a list-valued normal field is joined into a single normal-behavior string. Each item in the corresponding abnormal list is then converted into one rule entry. Scene-level rules additionally retain their prototype identifier; object and condition rules retain the detector category; and condition rules also retain the generated scene string. For example, the first abnormal item in the scene-level output above becomes:

Listing 9: Serialized scene-level rule entry after parsing.

```
{
  "level": "scene",
```

```
  "scene": "Public Square",
  "object_type": "",
  "normal_behavior": "People walking or standing casually\uFF1BBenches for
  sitting\uFF1BTrees providing shade\uFF1BOpen space for leisure activities",
  "potential_anomaly": "A sudden increase in the number of people",
  "text": "A sudden increase in the number of people",
  "confidence": 1.0,
  "prototype_id": 0
}
```


Similarly, the first condition-level item becomes:

Listing 10: Serialized condition-level rule entry after parsing.


```
{
  "level": "conditional",
  "scene": "Public Square",
  "object_type": "person",
  "normal_behavior": "",
  "potential_anomaly": "engaging in aggressive behavior towards others",
  "text": "engaging in aggressive behavior towards others",
  "confidence": 1.0
}
```


The value `confidence=1.0` is a storage default rather than a calibrated VLM confidence score and is not used as an additional learned weight. After construction, the normal and potential-anomaly strings are embedded by the frozen contrastive vision–language encoder and cached. Therefore, all online decisions are made by deterministic embedding similarity and rule routing; the generative VLM is not queried at test time.

C Analysis of Prototypes and Induced Rule Bank

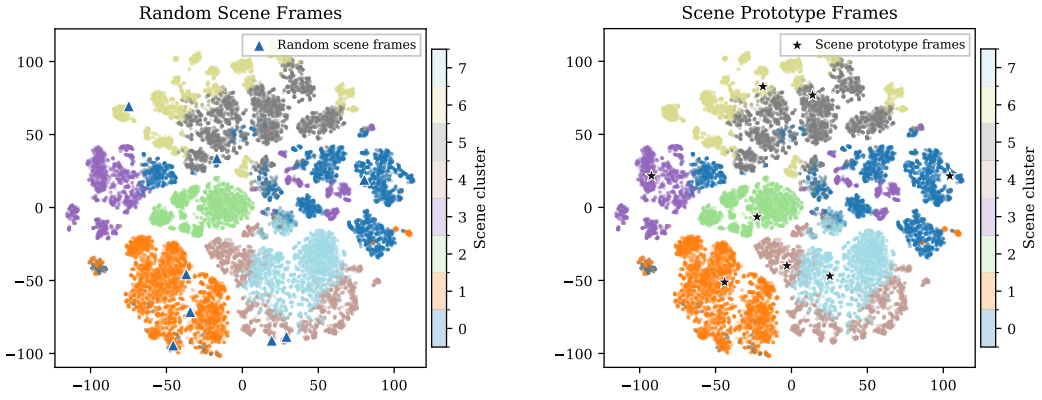
C.1 Prototype Representativeness Analysis

To further examine whether prototype selection provides a representative summary of normal visual patterns, we compare the scene prototypes with randomly selected normal frames using the same number of representative frames on ShanghaiTech. As shown in Fig. 4, all normal training frames are encoded by CLIP ViT-B/32 and projected into two dimensions using t-SNE for visualization, while the colors indicate the eight K-Means clusters obtained in the original feature space. With eight representative samples, the random selection shown in Fig. 4a covers only five of the eight clusters and repeatedly samples several clusters. In contrast, our prototype selection chooses the real training frame nearest to each cluster centroid and therefore covers all eight normal scene clusters, as shown in Fig. 4b.

We further quantify the representativeness in the original CLIP feature space using Coverage Error and Cluster Coverage. Over ten random seeds, random selection obtains a Coverage Error of 0.4784 ± 0.0282 and a Cluster Coverage of 0.6375 ± 0.1179, whereas prototype selection achieves 0.3709 and 1.0000, respectively. The lower coverage error and complete cluster coverage indicate that prototype-based selection provides a more comprehensive representation of recurring normal scene patterns under the same sample budget. Note that t-SNE is used only for qualitative visualization, while the quantitative coverage measures are computed in the original feature space.

C.2 Rule Bank Composition and Quality Analysis

We further examine the quality of the induced rule banks. We assess four complementary properties. *Semantic specificity* asks whether a rule describes an observable state or action rather than an image



(a) Random normal frames.

(b) Scene prototype frames.

Figure 4: Comparison between randomly selected normal scene frames and scene prototypes on ShanghaiTech. Background points denote CLIP embeddings of normal training frames projected into two dimensions using t-SNE, while colors indicate K-Means scene clusters obtained in the original feature space. (a) Eight randomly selected normal frames under the same sample budget, covering only a subset of the normal scene clusters. (b) Eight prototype frames selected as the real samples nearest to individual cluster centroids, covering all scene clusters.

identifier. *Format compliance* checks whether the parsed field contains plain natural language instead of a serialized nested object. *Redundancy* counts repeated rule tuples. Finally, *visual grounding* is inspected through the normal prototype crops supplied to the VLM. These metrics measure the cleanliness of the generated bank.

Table 4: Composition and deterministic quality analysis of the generated rule banks. "Duplicate" reports exact duplicate tuples in excess of the first occurrence. "Terse" denotes potential-anomaly fields containing at most two whitespace-delimited words.

Dataset	Total	Scene	Object	Condition	Image-index	Serialized dict.	Duplicate	Terse
ShanghaiTech	650	30	30	590	18 (2.77%)	7 (1.08%)	0 (0.00%)	15 (2.31%)
UCF-Crime	2,385	44	121	2,220	50 (2.10%)	13 (0.55%)	4 (0.17%)	69 (2.89%)
XD-Violence	1,917	39	98	1,780	40 (2.09%)	1 (0.05%)	12 (0.63%)	55 (2.87%)
Total	4,952	113	249	4,590	108 (2.18%)	21 (0.42%)	16 (0.32%)	139 (2.81%)

As shown in Table 4, the three banks contain 4,952 rules, of which 4,590 are condition-level rules. The strong concentration at the condition-level is expected because each retained scene–object pair can produce multiple anomaly hypotheses. Scene-level and object-level rules provide broader priors, whereas condition-level rules encode context-specific constraints. The counts also make clear that rule-bank size should not be read as the number of independent behaviors: several hypotheses may share a scene, object category, or normal description.

Most entries are well-formed natural-language hypotheses. For example, a ShanghaiTech scene-level rule describes a *Public Square*, normal activity as "people walking or standing casually," and a potential anomaly as "unusual behavior such as running or screaming." Such a rule is observable, tied to its context, and can be compared directly with video features.

Overall, the induced rule banks are largely well-formed, with generation artifacts accounting for only a small fraction of the entries. Among the 4,952 generated rules, 108 (2.18%) contain references to input positions rather than fully semantic descriptions, and 21 (0.42%) contain serialized dictionary-like text due to permissive parsing. Exact duplication is rare, with only 16 excess entries (0.32%). A conservative terse-text heuristic additionally flags 139 rules (2.81%), although concise expressions may still constitute valid semantic rules. Notably, all reported statistics are measured on the original, unfiltered rule banks used in the experiments, without manual editing or post-hoc rule selection.

D Qualitative Analysis of Inference Stage

We further analyze the inference stage of ProtoRule-VAD through representative examples. We first illustrate how hierarchical rules are retrieved and matched during inference, showing how normal and potential-anomaly evidence jointly determine the rule-violation score. We then present additional successful detections and representative failure cases to examine both the strengths and limitations of the proposed inference pipeline.

D.1 Rule-Based Inference Reasoning

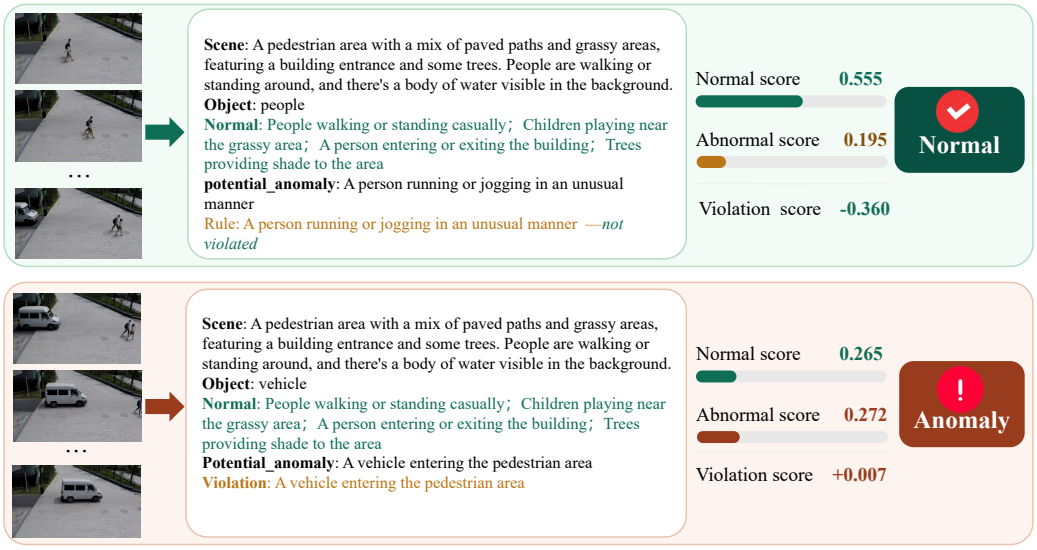
To provide further insight into the hierarchical rule-matching process, we present representative normal and anomalous examples from ShanghaiTech and UCF-Crime. For each test clip, ProtoRule-VAD retrieves context-compatible rules according to the inferred scene and detected object categories, and compares the visual representation with both normal and potential-anomaly descriptions. The corresponding rule-violation margin determines whether the semantic evidence favors normality or anomaly.

ShanghaiTech. On the ShanghaiTech dataset, we select a typical pedestrian-area scene to illustrate the reasoning process of the model. This scene contains roads, grass areas, building entrances, and pedestrian activities, so the normal rules mainly describe daily behaviors such as pedestrians walking, standing, entering buildings, and leaving buildings. For the normal clip in the figure5a, although the model retrieves a pedestrian-related potential anomaly rule, A person running or jogging in an unusual manner, this rule is not triggered by the current clip. Specifically, the matching score between the current clip and the normal-behavior rules is 0.555, which is higher than its matching score with the anomaly rule, 0.195. The resulting rule violation score is therefore -0.360, and the clip is classified as normal. This result shows that the model does not classify a clip as anomalous simply because a potential anomaly rule is retrieved. Instead, it further compares the semantic matching degree between the current clip and both normal and anomalous rules.

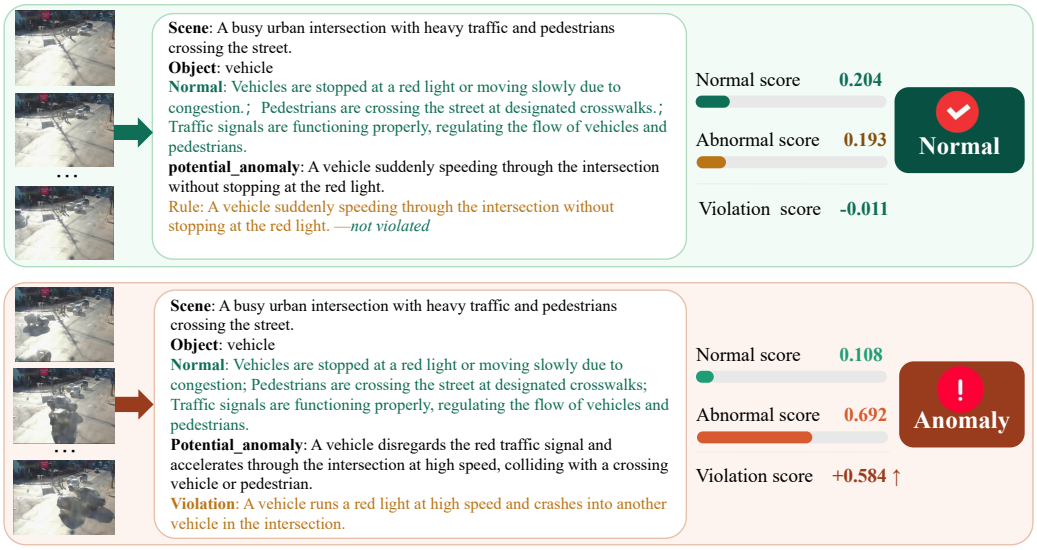
By contrast, the anomalous clip in the figure5a shows a vehicle entering a pedestrian area.For this clip, the model retrieves a vehicle-related potential anomaly rule according to the current scene and object category, A vehicle entering the pedestrian area. Since this behavior is inconsistent with normal activities in a pedestrian area, the matching score between the current clip and the anomaly rule is slightly higher than that with the normal rules, leading to a positive rule violation score of $+0.007$. The clip is therefore classified as anomalous. This example illustrates the ability of ProtoRule-VAD to identify scene-dependent anomalies. A vehicle is not necessarily anomalous by itself, but when it appears in an area mainly intended for pedestrian activities, it violates the normal behavioral constraints of that scene.

UCF-Crime. On the UCF-Crime dataset, we show a normal traffic clip and an anomalous accident clip from an urban road-intersection scene.The rules of this scene mainly focus on vehicles, traffic signals, and intersection-crossing behaviors. For the normal clip in the figure5b, the vehicles remain in a normal traffic state, such as waiting at a red light, moving slowly, or passing through the intersection according to traffic signals. Although the model retrieves a potential anomaly rule, A vehicle suddenly speeding through the intersection without stopping at the red light, the current clip does not violate this rule. Its normal-rule matching score is 0.204, which is higher than the anomaly-rule matching score of 0.193. The resulting rule violation score is therefore -0.011, and the model finally classifies the clip as normal.

For the anomalous clip in the figure5b, a vehicle rushes into the intersection at high speed and causes a collision. The retrieved potential anomaly rule explicitly describes a vehicle ignoring a red light, passing through the intersection at high speed, and colliding with other vehicles or pedestrians. The matching score between this clip and the anomaly rule reaches 0.692, which is significantly higher than the normal-rule matching score of 0.108. The resulting rule violation score is therefore $+0.584$, and the model finally classifies the clip as anomalous. This result shows that ProtoRule-VAD not only outputs an anomaly score, but also associates its anomaly decision with specific semantic rules, thereby providing a more intuitive reasoning basis.



(a) Qualitative analysis on ShanghaiTech.



(b) Qualitative analysis on UCF-Crime.

Figure 5: The inference process of the model on the ShanghaiTech and UCF-Crime datasets. For each test clip, the model first retrieves relevant hierarchical rules according to the scene and object category, including the scene description, normal-behavior rules, and potential-anomaly rules. It then computes the similarities between the current clip and both normal and anomalous rules, and obtains a rule violation score by subtracting the normal-rule similarity from the anomaly-rule similarity. A negative violation score indicates that the clip is more consistent with normal behavior and is therefore classified as normal, while a positive violation score indicates that the clip is closer to potential anomalous rules and is thus classified as anomalous.

D.2 Representative Successful Detections

Figures 6 provide additional successful examples on ShanghaiTech. Their ROC-AUC values are 0.983 and 0.968, respectively, and the high-score regions largely overlap the annotated anomaly intervals. In Figure 6a, window 160–184 obtains a scene-level violation margin of $+0.0217$ for

the hypothesis "unusual behavior from individuals (e.g., running, fighting)," while the dominant person-object margin reaches $+0.0621$. In Figure 6b, the retained window 160–184 receives both a positive scene margin ($+0.0134$) and a positive person-object margin ($+0.0479$). These cases illustrate the intended interaction between candidate prescreening, condition-guided rule routing, and hierarchical semantic matching: when the anomalous event is successfully retained and routed to compatible rules, scene-level and object-level evidence can reinforce one another.

D.3 Failure Cases and Error Sources

We further analyze representative failed predictions to identify the main error sources of the current pipeline.

Figure 7a contains ordinary pedestrians over frames 160–184, which are labeled normal. Nevertheless, this window receives a final anomaly score of 0.0590. The scene comparison itself is correct: its best violation margin is -0.0564, favoring the normal description. The largest person-object margin is instead $+0.0590$, and the maximum fusion used by the method propagates this value to the final score. This exposes a calibration mismatch across hierarchy levels: one weak positive object match can override consistent negative scene evidence. The same case contains an annotated event over frames 228–335. None of the selected windows overlaps this interval: the selected windows end at 216 before the event and resume at frame 400 after it. The event therefore receives zero score without reaching hierarchical rule matching. Visually, the relevant interaction is small and distant, so the motion/novelty prescreener ranks it below more salient normal motion. This is the principal recall limitation of sparse candidate evaluation. Adaptive candidate budgets, multi-scale motion features, or a low-cost fallback scan are plausible ways to improve coverage while retaining most of the computational advantage. We leave both calibration and adaptive candidate selection to future work to avoid introducing additional validation-dependent components into the reported system.

Figure 7b shows a vehicle traversing a pedestrian area during the annotated interval (frames 127–220). Unlike the previous case, window 192–216 is retained. However, it is routed to "a park or urban recreational area" and the closest scene-level anomaly hypothesis is "a person suddenly running at high speed without a clear reason." Its scene violation margin is -0.0203, the object branch supplies no positive margin, and the final score is zero. This case shows that candidate recall alone is insufficient: errors in scene assignment or object recognition restrict the conditional rules that can be considered. Soft routing to several scene prototypes and an open-set fallback rule would reduce the impact of a single incorrect route.

E Discussion and Limitations

ProtoRule-VAD still faces several limitations. Severe occlusion, low-resolution inputs, or substantial domain shifts may reduce the representativeness of the constructed prototypes and the reliability of the induced semantic rules. In addition, the current matching procedure mainly relies on short-range evidence within candidate segments and may therefore be less effective for anomalies that require long-term temporal context or causal dependencies across extended events. Future work could explore more robust prototype construction and long-range temporal reasoning to better handle challenging visual conditions and temporally extended anomalies.

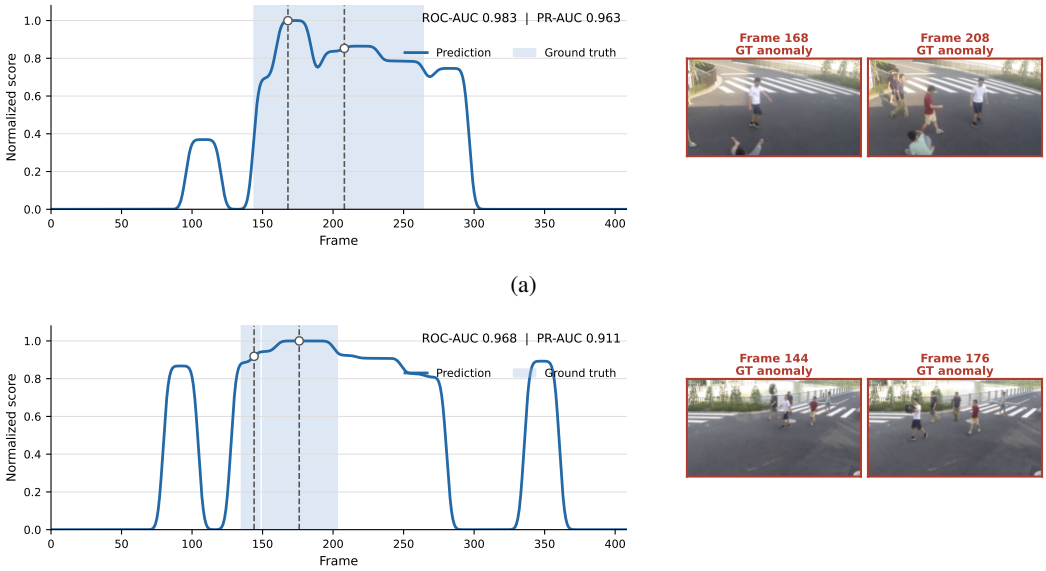


Figure 6: Successful detections on ShanghaiTech. Predicted scores align with the shaded ground-truth intervals. Curves are Gaussian-smoothed and normalized per video only for visualization; AUC values are computed from the saved, unsmoothed frame-level outputs.

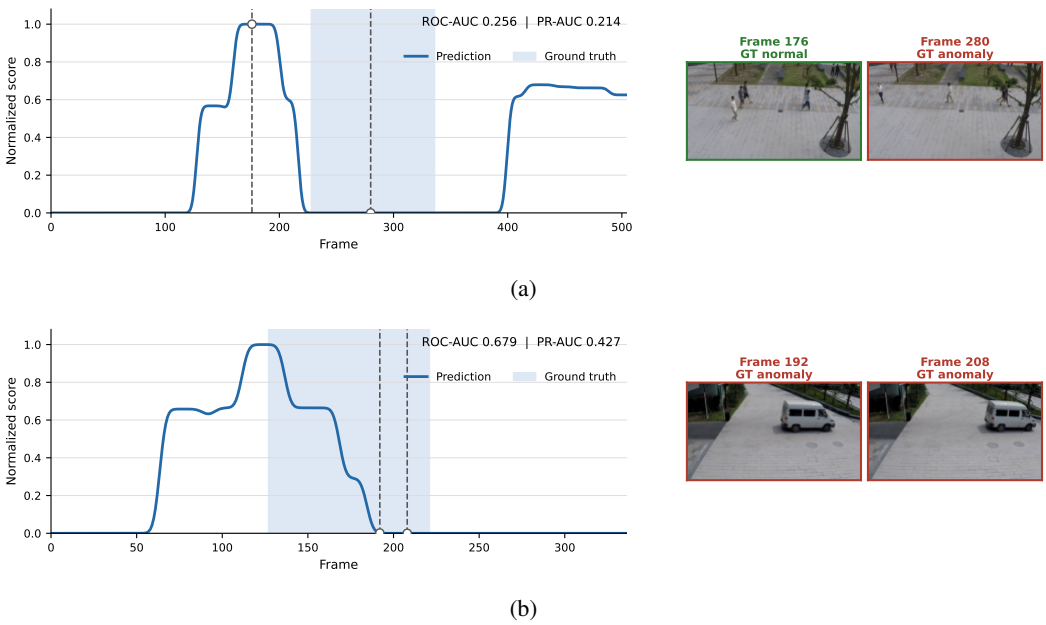


Figure 7: Failure cases on ShanghaiTech. The first row exhibits a high normal-region response and a complete candidate-gate miss of the annotated event. In the second row, the vehicle event is retained but receives zero score after incorrect semantic routing.