Follow the Rules: Reasoning for Video Anomaly Detection with Large Language Models

Yuchen Yang[1*], Kwonjoon Lee[2], Behzad Dariush[2], Yinzhi Cao[1], and
Shao-Yuan Lo[2]

[1] Johns Hopkins University
{yc.yang, yinzhi.cao}@jhu.edu
[2] Honda Research Institute USA
{kwonjoon_lee, bdariush, shao-yuan_lo}@honda-ri.com

Abstract. Video Anomaly Detection (VAD) is crucial for applications such as security surveillance and autonomous driving. However, existing VAD methods provide little rationale behind detection, hindering public trust in real-world deployments. In this paper, we approach VAD with a reasoning framework. Although Large Language Models (LLMs) have shown revolutionary reasoning ability, we find that their direct use falls short of VAD. Specifically, the implicit knowledge pre-trained in LLMs focuses on general context and thus may not apply to every specific real-world VAD scenario, leading to inflexibility and inaccuracy. To address this, we propose AnomalyRuler, a novel rule-based reasoning framework for VAD with LLMs. AnomalyRuler comprises two main stages: induction and deduction. In the induction stage, the LLM is fed with few-shot normal reference samples and then summarizes these normal patterns to induce a set of rules for detecting anomalies. The deduction stage follows the induced rules to spot anomalous frames in test videos. Additionally, we design rule aggregation, perception smoothing, and robust reasoning strategies to further enhance AnomalyRuler's robustness. AnomalyRuler is the first reasoning approach for the one-class VAD task, which requires only few-normal-shot prompting without the need for full-shot training, thereby enabling fast adaption to various VAD scenarios. Comprehensive experiments across four VAD benchmarks demonstrate AnomalyRuler's state-of-the-art detection performance and reasoning ability. AnomalyRuler is open-source and available at: https://github.com/Yuchen413/AnomalyRuler

1 Introduction

Video Anomaly Detection (VAD) aims to identify anomalous activities, which are infrequent or unexpected in surveillance videos. It has a wide range of practical applications, including security (e.g., violence), autonomous driving (e.g., traffic accidents), etc. VAD is a challenging problem since anomalies are rare and long-tailed in real life, leading to a lack of large-scale representative anomaly data.

---

[*] This work was mostly done when Y. Yang was an intern at HRI-USA.
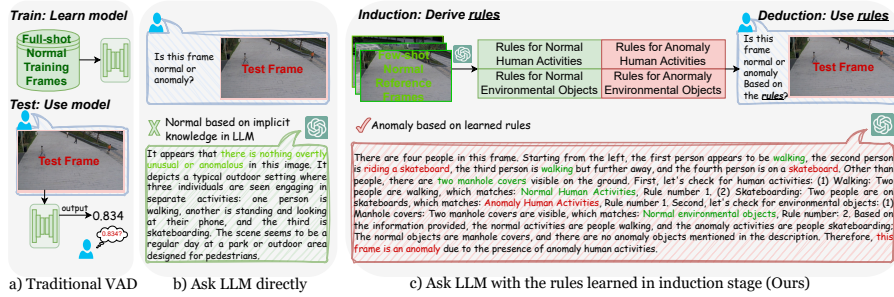
Fig. 1: Comparison of one-class VAD approaches. In this specific safety application example, only "walking" is normal. The test frame contains "skateboarding", so it is abnormal. (a) Traditional methods require full-shot training and only output anomaly scores, lacking reasoning. (b) Direct LLM use may not align with specific VAD needs. Here GPT-4V mistakenly treats "skateboarding" as normal. (c) Our AnomalyRuler has induction and deduction stages. It derives rules from few-shot normal reference frames to detect anomalies, correctly identifying "skateboarding" as an anomaly.

Hence, the one-class VAD (a.k.a. unsupervised VAD) paradigm [16,36,43,45,53] is preferred, as it assumes that only the more accessible normal data are available for training. Most existing one-class VAD methods learn to model normal patterns via self-supervised pretext tasks, such as frame reconstruction [16,25,27,36,45, 53,54] and frame order classification [14,43,49]. Despite good performance, these traditional methods can only output anomaly scores, providing little rationale behind their detection results (see Fig. 1a). This hinders them from earning public trust when deployed in real-world products.

We approach the VAD task with a reasoning framework toward a trustworthy system, which is less explored in the literature. An intuitive way is to incorporate the emergent Large Language Models (LLMs) [1,7,19,38,39,47], which have shown revolutionary capability in various reasoning tasks. Still, we find that their direct use falls short of performing VAD. Specifically, the implicit knowledge pre-trained in LLMs focuses on general context, meaning that it may not always align with specific real-world VAD applications. In other words, there is a mismatch between an LLM's understanding of anomalies and the anomaly definitions required for certain scenarios. For example, the GPT-4V [1] typically treats "skateboarding" as a normal activity, whereas certain safety applications need to define it as an anomaly, such as within a restricted campus (see Fig. 1b). However, injecting such specific knowledge by fine-tuning LLMs for each application is costly. This highlights the necessity for a flexible prompting approach that steers LLMs' reasoning strengths to different uses of VAD.

To arrive at such a solution, we revisit the fundamental process of the scientific method [4] emphasizing reasoning, which involves drawing conclusions in a rigorous manner [41]. Our motivation stems from two types of reasoning: inductive reasoning, which infers generic principles from given observations, and deductive reasoning, which derives conclusions based on given premises. In this

paper, we propose AnomalyRuler, a new VAD framework based on reasoning with LLMs. AnomalyRuler consists of an induction stage and a deduction stage as shown in Fig. 1c. In the induction stage, the LLM is fed with visual descriptions of few-shot normal samples as references to derive a set of rules for determining normality. Here we employ a Vision-Language Model (VLM) [23, 51] to generate the description for each input video frame. Next, the LLM derives a set of rules for detecting anomalies by contrasting the rules for normality. The deduction, which is also an inference stage, follows the induced rules to identify anomalous frames in test video sequences. Additionally, in response to potential perception and reasoning errors by the VLM and LLM, we design strategies including rule aggregation via the randomized smoothing [10] for rule induction error mitigation, perception smoothing via the proposed Exponential Majority Smoothing for perception error reduction together with temporal consistency enhancement, and robust reasoning via a recheck mechanism for reliable reasoning output. These strategies are integrated into the AnomalyRuler pipeline to further enhance its detection robustness.

Apart from equipping VAD with reasoning ability, AnomalyRuler offers several advantages. First, AnomalyRuler is a novel *few-normal-shot prompting* approach that utilizes only a few normal samples from a training set as references to derive the rules for VAD. This avoids the need for expensive full-shot training or fine-tuning of the entire training set, as required by traditional one-class VAD methods. Importantly, it enables efficient adaption by redirecting LLM's implicit knowledge to different specific VAD applications through just a few normal reference samples. Second, AnomalyRuler shows strong domain adaptability across datasets, as the language provides consistent descriptions across different visual domains, e.g., "walking" over visual data variance. This allows the application of induced rules to datasets with similar scenarios but distinct visual appearances. Furthermore, AnomalyRuler is a generic framework that is complementary to VLM and LLM backbones. It accommodates both closed-source models such as the GPT family [1, 39] and open-source alternatives such as Mistral [19]. To the best of our knowledge, the proposed AnomalyRuler is the first reasoning approach for the one-class VAD problem. Extensive experiments on four VAD datasets demonstrate AnomalyRuler's state-of-the-art performance, reasoning ability, and domain adaptability.

In summary, this paper has three main contributions. (1) We propose a novel rule-based reasoning framework for VAD with LLMs, namely AnomalyRuler. To the best of our knowledge, it is the first reasoning approach for one-class VAD. (2) The proposed AnomalyRuler is a novel few-normal-shot prompting approach that eliminates the need for expensive full-shot tuning and enables fast adaption to various VAD scenarios. (3) We propose rule aggregation, perception smoothing, and robust reasoning strategies for AnomalyRuler to enhance its robustness, leading to state-of-the-art detection performance, reasoning ability, and domain adaptability.

2   Related Work

Video Anomaly Detection. VAD is a challenging task since anomaly data are scarce and long-tailed. Therefore, researchers often focus on the one-class VAD (a.k.a. unsupervised VAD) paradigm [14, 16, 18, 25, 27, 36, 43, 45, 49, 53, 54], which uses only normal data during training. Most one-class methods learn to model normal patterns via self-supervised pretext tasks, based on the assumption that the model would obtain poor pretext task performance on anomaly data. Reconstruction-based methods [16, 25, 27, 36, 45, 53, 54] employ generative models such as auto-encoders and diffusion models to perform frame reconstruction or frame prediction as pretext tasks. Distance-based [14, 43, 49] methods use classifiers to perform pretext tasks such as frame order classification. These traditional methods can only output anomaly scores, providing little rationale behind their detection. Several recent studies explore utilizing VLMs or LLMs in anomaly detection. Elhafsi et al. [12] analyze semantic anomalies with an object detector [33] an LLM [7] in driving scenes. However, it relies on predefined concepts of normality and anomaly, which limits its adaption to different scenarios and cannot handle long-tailed undefined anomalies. Moreover, this method has not been evaluated on standard VAD benchmarks [2, 22, 24, 28]. Cao et al. [8] explore the use of GPT-4V for anomaly detection, but their direct use may fall into the misalignment between GPT-4V's implicit knowledge and specific VAD needs, as discussed. Gu et al. [15] adopt a large VLM for anomaly detection, but it focuses on industrial images. Despite supporting dialogues, this method can only describe anomalies rather than explain the rationales behind its detection. Lv et al. [31] equip video-based LLMs in the VAD framework to provide detection explanations. It involves three-phase training to fine-tune the heavy video-based LLMs. Besides, it focuses on weakly-supervised VAD, a relaxed paradigm that requires training with anomaly data and labels. Different from these works, our AnomalyRuler provides rule-based reasoning via efficient few-normal-shot prompting and enables fast adaption to different VAD scenarios.

Large Language Models. LLMs [1, 7, 19, 38, 39, 46, 47] have achieved significant success in natural language processing and are recently being explored for computer vision problems. Recent advances, such as the GPT family [1, 7, 38, 39], the LLaMA family [46, 47], and Mistral [19], have shown remarkable capabilities in understanding and generating human language. On the other hand, large VLMs [1, 21, 23, 34, 44, 51, 57, 58, 60] have shown promise in bridging the vision and language domains. BLIP-2 [21] leverages Q-Former to integrate visual features into a language model. LLaVA [23] introduces a visual instruction tuning method for visual and language understanding. CogVLM [51] trains a visual expert module to improve large VLM's vision ability. Video-LLaMA [57] extends LLMs to understand video data. These models' parametric knowledge is trained for general purposes and thus may not apply to every VAD application. Recent studies explore prompting methods to exploit LLMs' reasoning ability. Chain-of-Thought (CoT) [11, 52] guides LLMs to solve complex problems via multiple smaller and manageable intermediate steps. Least-to-Most (LtM) [20, 59] decomposes a complex problem into multiple simpler sub-problems and solves them in sequence.
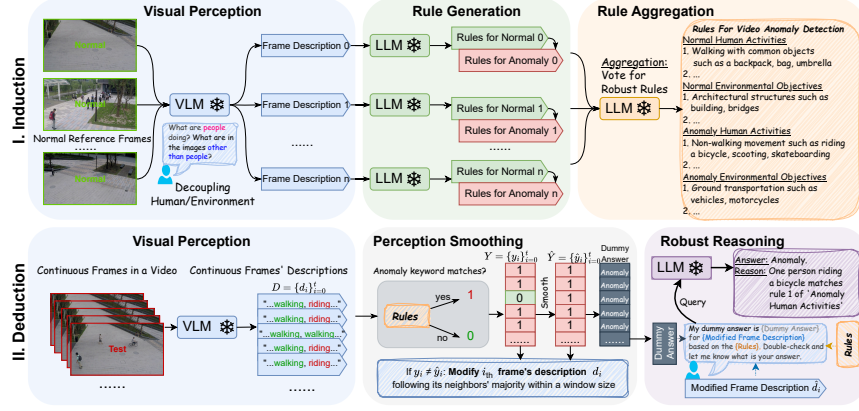
Fig. 2: The AnomalyRuler pipeline consists of two main stages: induction and deduction. The induction stage involves: i) visual perception transfers normal reference frames to text descriptions; ii) rule generation derives rules based on these descriptions to determine normality and anomaly; iii) rule aggregation employs a voting mechanism to mitigate errors in rules. The deduction stage involves: i) visual perception transfers continuous frames to descriptions; ii) perception smoothing adjusts these descriptions considering temporal consistency to ensure neighboring frames share similar characteristics; iii) robust reasoning rechecks the previous dummy answers and outputs reasoning.

Hypotheses-to-Theories (HtT) [61] learns a rule library for reasoning from labeled training data in a supervised manner. However, a reasoning approach for the VAD task in the one-class paradigm is not well-explored.

3 Induction

The induction stage aims to derive a set of rules from a few normal reference frames for performing VAD. The top part of Fig. 2 shows the three modules in the induction pipeline. The visual perception module utilizes a VLM which takes a few normal reference frames as inputs and outputs frame descriptions. The rule generation module uses an LLM to generate rules based on these descriptions. The rule aggregation module employs a voting mechanism to mitigate the errors from rule generation. In the following sections, we discuss each module and the strategies applied in detail.

3.1 Visual Perception

We design the visual perception module as the initial step in our pipeline. This module utilizes a VLM to convert video frames into text descriptions. We define $F_{\mathrm{normal}} = \{f_{\mathrm{normal}_0}, \ldots, f_{\mathrm{normal}_n}\}$ as the few-normal-shot reference frames, with each frame $f_{\mathrm{normal}_i} \in F_{\mathrm{normal}}$ randomly chosen from the training set. This module outputs the text description of each normal reference frame:

$D_{\text{normal}} = \{\text{VLM}(f_{\text{normal}_i}, p_v) \mid f_{\text{normal}_i} \in F_{\text{normal}}\}$, with p_v as the prompt "What are people doing? What are in the images other than people?". Instead of directly asking "What are in the image?", we design p_v to separate humans and the environment with the following advantages. First, it enhances perception precision by directing the model's attention to specific aspects of the scene, ensuring that no details are overlooked. Second, it simplifies the following rule generation module by dividing the task into two subproblems [20], i.e., rules for human activities and rules for environmental objects. We denote this strategy as Human and Environment.

3.2   Rule Generation

With the text descriptions from normal reference frames D_{normal}, we design a Rule Generation module that uses a frozen LLM to generate rules (denoted as R). In formal terms, $R = \{\text{LLM}(d_{\text{normal}_i}, p_g) \mid d_{\text{normal}_i} \in D_{\text{normal}}\}$, where p_g is the prompt detailed in Appendix A.2. We craft p_g with three strategies to guide the LLM in gradually deriving rules from the observed normal patterns:

Normal and Anomaly. The prompt p_g guides the LLM to perform *contrast*, which first induces rules for normal based on D_{normal}, which are assumed to be ground-truth normal. Then, it generates rules for anomalies by contrasting them with the rules for normal. For instance, if "walking" is a common pattern in D_{normal}, it becomes a normal rule, and then "non-walking movement" will be included in the rules for anomaly. This strategy sets a clear boundary between normal and anomaly without access to anomaly frames.

Abstract and Concrete. The prompt p_g helps the LLM to perform *analogy*, which starts from an abstract concept and then effectively generalizes to more concrete examples. Taking the same "walking" example, the definition of a normal rule is now expanded to "walking, whether alone or with others." Consequently, the anomaly rule evolves to include specific non-walking movements, i.e., "non-walking movement, such as riding a bicycle, scooting, or skateboarding." This strategy clarifies the rules with detailed examples and enables the LLM to use analogy for reasoning without exhaustively covering every potential scenario.

Human and Environment. This strategy is inherited from the Visual Perception module. The prompt p_g leads the LLM to pay attention separately to environmental elements (e.g., vehicles or scene factors) and human activities, separately. This enriches the rule set for VAD tasks, where anomalies often arise from interactions between humans and their environment.

These strategies align with the spirit of CoT [52] yet are further refined for the VAD task. The ablation study in Section 5.4 demonstrates their effectiveness.

3.3   Rule Aggregation

The rule aggregation module uses a LLM as an aggregator with a voting mechanism to combine n sets of rules (i.e., R) generated independently from n randomly chosen normal reference frames into one set of robust rules, $R_{\text{robust}} = \text{LLM}(R, p_a)$.

This module aims to mitigate errors from previous stages, such as the visual perception module's potential misinterpretation of "walking" as "skateboarding", leading to incorrect rules. The aggregation process filters out uncommon elements by retaining rule elements consistently present across the n sets. The prompt p_a for the LLM to achieve this is detailed in Appendix A.2. This strategy is based on the assumption of *randomize smoothing* [10], where errors may occur on a single input but are less likely to consistently occur across multiple randomly sampled inputs. Therefore, by aggregating these outputs, AnomalyRuler generates rules more resilient to individual errors. The hyperparameter n can be treated as the number of batches. For simplicity, previous discussions assume that each batch has only one frame, i.e., $m = 1$. Here we define m as the number of normal reference frames per batch, i.e., batch size. We show the effectiveness of the rule aggregation and provide an ablation on different n and m values in Section 5.4.

4 Deduction

After the induction stage derives a set of robust rules, the deduction stage follows these rules to perform VAD. The bottom part of Fig. 2 illustrates the deduction stage, which aims to precisely perceive each frame of videos and then use the LLM to reason if they are normal or abnormal based on the rules. To achieve this goal, we design three modules. First, the visual perception module works similarly as described in the induction stage. However, instead of taking the few-normal-shot reference frames, the deduction processes continuous frames from each test video and outputs a series of frame descriptions $D = \{d_0, d_1, \ldots, d_t\}$. Second, the perception smoothing module reduces errors with the proposed Exponential Majority Smoothing. This step alone can provide preliminary detection results, referred to as AnomalyRuler-base. Third, the robust reasoning module utilizes an LLM to recheck the preliminary detection results against the rules and perform reasoning. The perception smoothing and robust reasoning modules are elaborated in the following sections.

4.1 Perception Smoothing

As we discussed in Section 3.3, visual perception errors would happen in the induction stage, and this concern extends to the deduction stage as well. To address this challenge, we propose a novel mechanism named Exponential Majority Smoothing. This mechanism mitigates the errors by considering temporal consistency in videos, i.e., movements are continuous and should exhibit consistent patterns over time. We utilize the results of this smoothing to guide the correction of frame descriptions, enhancing AnomalyRuler's robustness to errors. There are four key steps:

Initial Anomaly Matching. For the continuous frame descriptions $D = \{d_0, d_1, \ldots, d_t\}$, AnomalyRuler first match anomaly keywords K found within the anomaly rules from the induction stage (see details in Appendix A.2), and assigns d_i with label y_i where $i \in [0, t]$, represents the predicted label. Formally,

we have $y_i = 1$ if $\exists k \in K \subseteq d_i$, indicating an anomaly triggered by keywords such as ing-verb "riding" or "running". Otherwise, $y_i = 0$ indicates the normal. We denote the initial matching predictions as $Y = \{y_0, y_1, \ldots, y_t\}$.

Exponential Majority Smoothing. We propose an approach that combines Exponential Moving Average (EMA) and Majority Vote. This approach is designed to enhance the continuity in human or object movements by adjusting the predictions to reflect the most common state within a specified window. The final smoothed predictions are denoted as $\hat{Y} = \{\hat{y}_0, \hat{y}_1, \ldots, \hat{y}_t\}$, where each $\hat{y}_i$ is either 1 or 0. Formally, we have:

- Step I: EMA. For original prediction y_t, the EMA value s_t is computed as $s_t = \frac{\sum_{i=0}^{t}(1-\alpha)^{t-i} y_i}{\sum_{i=0}^{t}(1-\alpha)^i}$. We denote α as the parameter that influences the weighting of data points in the EMA calculation.

- Step II: Majority Vote. The idea is to apply a majority vote to smooth the prediction within a window centered at each EMA value s_i with a padding size p. This means that for each s_i, we consider its neighboring EMA values within the window and determine the smoothed prediction $\hat{y}_i$ based on the majority of these values being above or below a threshold τ. We define this threshold as the mean of all EMA values: $\tau = \frac{1}{t}\sum_{i=1}^{t} s_i$. Formally, the smoothed prediction $\hat{y}_i$ is determined as:

$$\hat{y}_i = \begin{cases} 1 & \text{if } \sum_{j=\max(1,i-p)}^{\min(i+p,t)} \mathbb{1}(s_j > \tau) > \frac{\min(i+p,t)-\max(1,i-p)+1}{2} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $\mathbb{1}(\cdot)$ denotes the indicator function and the window size is adaptively defined as $\min(i+p,t) - \max(1,i-p) + 1$ ensuring that the window does not extend beyond the boundaries determined by the range from $\max(1, i-p)$ to $\min(i+p,t)$.

Anomaly Score. Given that $\hat{Y}$ represents the initial detection results of AnomalyRuler, we can further assess these by calculating an anomaly score through a secondary EMA. Specifically, the *anomaly scores*, denoted as $A = \{a_0, a_1, \ldots, a_t\}$, where a_t is:

$$a_t = \frac{\sum_{i=0}^{t}(1-\tau)^{t-i}\hat{y}_i}{\sum_{i=0}^{t}(1-\tau)^i} \quad (2)$$

We denote the above procedure AnomalyRuler-base as a baseline of our method, which provides a *dummy answer*, i.e., "Anomaly" if $\hat{y}_i = 1$ otherwise "Normal", with an anomaly score that is comparable with the state-of-the-art VAD methods [3, 25, 35, 43]. Subsequently, AnomalyRuler utilizes the dummy answer in the robust reasoning module for further analysis.

Description Modification. In this step, AnomalyRuler modifies the description D comparing Y and $\hat{Y}$ and outputs the modified $\hat{D}$. If $y_i = 0$ while $\hat{y}_i = 1$, indicating a false negative in the perception module, AnomalyRuler corrects d_i by adding "There is a person $\{k\}$.", where $k \in K$ is the most frequent anomaly keyword within the window size w. Conversely, if $y_i = 1$ while $\hat{y}_i = 0$, indicating a false positive in the perception module, so AnomalyRuler modifies d_i by removing parts of the description that contain the anomaly keyword k.

4.2   Robust Reasoning

In the robust reasoning module, AnomalyRuler utilizes an LLM to achieve the reasoning task for VAD, with the robust rule R_{robust} derived from the induction stage as the context. The LLM is fed with each frame's modified description $\hat{d}_i$ with its dummy answer, i.e., either "Anomaly" or "Normal" generated from AnomalyRuler-base. We denote the output of robust reasoning as $Y^* = \{\mathrm{LLM}(\hat{d}_i, \hat{y}_i, R_{\mathrm{robust}}, p_r) \mid \hat{d}_i \in \hat{D}, \hat{y}_i \in \hat{Y}\}$. To ensure reliable results, the prompt p_r, detailed in Appendix A.2, guides the LLM to recheck whether the dummy answer $\hat{y}_i$ matches the description $\hat{d}i$ according to R_{robust}. This validation step, instead of directly asking the LLM to analyze $\hat{d}_i$, improves decision-making by using the dummy answer as a hint. This approach helps AnomalyRuler reduce missed anomalies (false negatives) and ensures that its reasoning is more closely aligned with the rules. Additionally, to compare AnomalyRuler with the state-of-the-art approaches based on thresholding anomaly scores, we apply Equation (2) with replacing $\hat{y}_i$ by $y_i^* \in Y^*$ to output anomaly scores.

5   Experiments

This section compares AnomalyRuler with LLM-based baselines and state-of-the-art methods in terms of both detection and reasoning abilities. We also conduct an ablation study on each module within AnomalyRuler to evaluate their contributions. Examples of complete prompts, derived rules, and outputs are illustrated in Appendix A.2.

5.1   Experimental Setup

Datasets. We evaluate our method on four VAD benchmark datasets. (1) UCSD Ped2 (Ped2) [22]: A single-scene dataset captured in pedestrian walkways with over 4,500 frames of videos, including anomalies such as skating and biking. (2) CUHK Avenue (Ave) [28]: A single-scene dataset captured in the CUHK campus avenue with over 30,000 frames of videos, including anomalies such as running and biking. (3) ShanghaiTech (ShT) [24]: A challenging dataset that contains 13 campus scenes with over 317,000 frames of videos, containing anomalies such as biking, fighting, and vehicles in pedestrian areas. (4) UBnormal (UB) [2]: An open-set virtual dataset generated by the Cinema4D software, which contains 29 scenes with over 236,000 frames of videos. For each dataset, we use the default training and test sets that adhere to the one-class setting. The normal reference frames used by AnomalyRuler are randomly sampled from the normal training set. The methods are evaluated on the entire test set if not otherwise specified.
Evaluation Metrics. Following the common practice, we use the Area Under the receiver operating characteristic Curve (AUC) as the main detection performance metric. To compare with LLM-based methods that cannot output anomaly scores, we use the accuracy, precision, and recall metrics. Besides, we adopt the Doubly-Right metric [32] to evaluate reasoning ability. All the metrics are calculated with frame-level ground truth labels.

Table 1: Detection performance with accuracy, precision, and recall (%) compared with different VAD with LLM methods on the ShT dataset.

Method	Accuracy	Precision	Recall
Ask LLM Directly	52.1	97.1	6.2
Ask LLM with Elhafsi et al. [12]	58.4	**97.9**	15.2
Ask Video-based LLM Directly	54.7	85.4	8.5
AnomalyRuler	**81.8**	90.2	**64.3**

Implementation Details. We implement our method, AnomalyRuler, using PyTorch [37]. If not otherwise specified, we employ CogVLM-17B [51] as the VLM for visual perception, GPT-4-1106-Preview [1] as the LLM for induction, and the open-source Mistral-7B-Instruct-v0.2 [19] as the LLM for deduction (i.e., inference) due to using GPTs on entire test sets is too costly. We discuss other VLMs/LLMs choices in Appendix A.4. The default hyperparameters of AnomalyRuler are set as follows: The number of batches in rule aggregation $n = 10$, the number of normal reference frames per batch $m = 1$, the padding size $p = 5$ in majority vote, and the weighting parameter $\alpha = 0.33$ in EMA.

5.2   Comparison with LLM-based Baselines

Reasoning for one-class VAD using LLMs is not well-explored. To demonstrate AnomalyRuler's superiority over the direct LLM use, we build asking LLM/Video-based LLM directly as baselines and also adapt related works [8, 12] to our target problem as baselines. At test time, let us denote test video frames as $F = \{f_1, f_2, \ldots, f_t\}$. We elaborate on our four baselines as follows. (1) Ask LLM Directly: $\{\text{LLM}(d_i, p) \mid d_i \in D\}$, where the LLM is Mistral-7B, D is F's frame descriptions generated by CogVLM, and p is "Is this frame description anomaly or normal?" (2) Ask LLM with Elhafsi et al. [12]: $\{\text{LLM}(d_i, p) \mid d_i \in D\}$, where the LLM is Mistral-7B, D is F's frame descriptions generated by CogVLM, and p is [12]'s prompts and predefined concepts of normality/anomaly. (3) Ask Video-based LLMs Directly: $\{\text{Video-based LLM}(c_i, p) \mid c_i \in C\}$, where p is "Is this clip anomaly or normal?" We use Video-LLaMA [57] as the Video-based LLM, which performs clip-wise inference. Each video clip c_i consists of consecutive frames in F with the same label. (4) Ask GPT-4V with Cao et al. [8]: $\{\text{GPT-4V}(f_i, p) \mid f_i \in F\}$, where p is [8]'s prompts. As a large VLM, GPT-4V directly takes frames as inputs.

Detection Performance. Table 1 compares the accuracy, precision, and recall on the ShT dataset. Overall, AnomalyRuler achieves significant improvements with an average increase of 26.2% in accuracy and 54.3% in recall. Such improvements are attributed to the reasoning based on the rules generated in the induction stage. In contrast, the baselines tend to predict most samples as normal based on the implicit knowledge pre-trained in LLMs, resulting in very low recall and accuracy close to a random guess. Their relatively high precision is due to that they rarely predict anomalies, leading to fewer false positives.

Table 2: Reasoning performance with the Doubly-Right metric: {RR, RW, WR, WW} (%) on 100 (limited by GPT-4's query capacity) randomly selected frames from the ShT test set. We evaluate cases with visual perception errors (w. Perception Errors) and with manually corrected visual perception (w/o. Perception Errors).

Method	w. Perception Errors				w/o. Perception Errors			
	RR	RW	WR	WW	RR	RW	WR	WW
Ask GPT-4 Directly	57	4	15	24	73	3	0	24
Ask GPT-4 with Elhafsi et al. [12]	60	3	15	22	76	2	0	22
Ask GPT-4V with Cao et al. [8]	74	2	7	17	81	2	0	17
AnomalyRuler	**83**	1	15	1	**99**	0	0	1

Reasoning Performance. The reasoning performance is evaluated using the Doubly-Right metric [32]: {RR, RW, WR, WW} (%), where RR denotes Right detection with Right reasoning, RW denotes Right detection with Wrong reasoning, WR denotes Right detection with Wrong reasoning, and WW denotes Wrong detection with Wrong reasoning. We desire a high accuracy of RR (the best is 100%) and low percentages of RW, WR and WW (the best is 0%). Since {RW, WR, WW} may be caused by visual perception errors rather than reasoning errors, we also consider the case with manually corrected visual perception to exclusively evaluate each method's reasoning ability, i.e., w. Perception Errors vs. w/o. Perception Errors in Table 2.

Due to the lack of benchmarks for evaluating reasoning for VAD, we create a dataset consisting of 100 randomly selected frames from the ShT test set, with an equal split of 50 normal and 50 abnormal frames. For each frame, we offer four choices: one normal and three anomalies, where only one choice with the matched rules is labeled as RR, while the other choices correspond to RW, WR or WW. Details and examples of this dataset are illustrated in Appendix A.3. Since the 100 randomly selected frames are not consecutive, here AnomalyRuler's perception smoothing is not used.

Table 2 shows the evaluation results. With perception errors, AnomalyRuler outperforms the baselines by 10% to 27% RR, and it achieves a very low WW of 1% compared to the 17% WW of the second best Ask GPT-4V with Cao et al. [8]. Without perception errors, AnomalyRuler's RR jumps to 99%. These results demonstrate AnomalyRuler's superiority over the GPT-4(V) baselines and its great ability to make correct detection along with correct reasoning.

5.3 Comparison with State-of-the-Art Methods

This section compares AnomalyRuler with 15 state-of-the-art one-class VAD methods across four datasets, evaluating their detection performance and domain adaptability. The performance values of these methods are sourced from their respective original papers.

Detection Performance. Table 3 shows the effectiveness of AnomalyRuler. There are three main observations. First, AnomalyRuler, even with its basic version AnomalyRuler-base, outperforms all the Image-Only competitors, which

Table 3: AUC (%) compared with different one-class VAD methods. "Image Only" methods only rely on image features. In contrast, others employ additional features such as bounding boxes from object detectors or 3D features from action recognition networks. "Training" indicates the methods that need a full-shot training process.

Method	Venue	Image Only	Training	Ped2	Ave	ShT	UB
MNAD [36]	CVPR-20	✓	✓	97.0	88.5	70.5	-
rGAN [29]	ECCV-20	✓	✓	96.2	85.8	77.9	-
CDAE [9]	ECCV-20	✓	✓	96.5	86.0	73.3	-
MPN [30]	CVPR-21	✓	✓	96.9	89.5	73.8	-
NGOF [50]	CVPR-21	✗	✓	94.2	88.4	75.3	-
HF2 [25]	ICCV-21	✗	✓	**99.2**	91.1	76.2	-
BAF [14]	TPAMI-21	✗	✓	98.7	92.3	82.7	59.3
GCL [56]	CVPR-22	✗	✓	-	-	79.6	-
S3R [53]	ECCV-22	✗	✓	-	-	80.5	-
SSL [49]	ECCV-22	✗	✓	99.0	**92.2**	84.3	-
zxVAD [3]	WACV-23	✗	✓	96.9	-	71.6	-
HSC [45]	CVPR-23	✗	✓	98.1	93.7	83.4	-
FPDM [54]	ICCV-23	✓	✓	-	90.1	78.6	62.7
SLM [43]	ICCV-23	✓	✓	97.6	90.9	78.8	-
STG-NF [18]	ICCV-23	✗	✓	-	-	**85.9**	71.8
AnomalyRuler-base	-	✓	✗	96.5	82.2	84.6	69.8
AnomalyRuler	-	✓	✗	97.9	89.7	85.2	**71.9**

Table 4: AUC (%) compared with different cross-domain VAD methods. We follow the compared works to use ShT as the source domain dataset for other target datasets.

Method	Venue	Image Only	Training	Ped2	Ave	ShT[1]	UB
rGAN [29]	ECCV-20	✓	✓	81.9	71.4	77.9	-
MPN [30]	CVPR-21	✓	✓	84.7	74.1	73.8	-
zxVAD [3]	WACV-23	✗	✓	95.7	**82.2**	71.6	-
AnomalyRuler-base	-	✓	✗	**97.4**	81.6	**83.5**	**65.4**

[1] AnomalyRuler employs UB as the source domain when ShT serves as the target domain. The competitors have no cross-domain evaluation on ShT, so we report their same-domain results.

also do not use any additional features (e.g., bounding boxes from object detectors or 3D features from action recognition networks), on the challenging ShT and UB datasets. This suggests that our rule-based reasoning benefits the challenging one-class VAD task. Second, for Ped2 and Ave, AnomalyRuler performs on par with the Image-Only methods. This is achieved without any tuning, meaning that our few-normal-shot prompting approach is as effective as the costly full-shot training on these benchmarks. Third, AnomalyRuler outperforms AnomalyRuler-base, indicating that the robust reasoning module improves performance further.

Domain Adaptability. Domain adaptation considers the scenario that the source domain (i.e., training/induction) dataset differs from the target domain (i.e., testing/deduction) dataset [13, 26, 48]. We compare AnomalyRuler with three state-of-the-art VAD methods that claim their domain adaptation ability [3, 29, 30]. We follow the compared works to use ShT as the source domain dataset for other target datasets. As shown in Table 4, AnomalyRuler achieves the highest AUC on Ped2, ShT and UB, outperforming with an average of 9.88%. While AnomalyRuler trails zxVAD [3] by 0.6%, it is still higher than the others with an average of 8.85%. The results indicate that AnomalyRuler has better

domain adaptability across different datasets. This advantage is due to that the language provides consistent descriptions across different visual domains, which allows the application of induced rules to datasets with similar anomaly scenarios but distinct visual appearances. In contrast, traditional methods extract high-dimensional visual features that are sensitive to visual appearances, thereby struggling to transfer their knowledge across datasets.

5.4 Ablation Study

In this section, we look into how the proposed strategies affect AnomalyRuler. We investigate two aspects: rule quantity (i.e., the number of induced rules) and rule quality (i.e., their resulting performance). Regarding this, we evaluate variants of AnomalyRuler-base on the ShT dataset.

Ablation on Strategies. Table 5 shows the effects of removing individual strategies compared to using all strategies. In terms of rule quantity, removing Human and Environment or Normal and Anomaly significantly reduces rules by 47.6% and 82.4%, respectively. This reduction is due to not separating the rules for humans and the environment halves the number of rules. Moreover, without deriving anomaly rules from normal rules, we only have a limited set of normal rules. Removing Abstract and Concrete or Rule Aggregation slightly increases the number of rules, as the former merges rules within the same categories and the latter removes incorrect rules. Perception Smoothing does not affect rule quantity since it is used in the deduction stage. In terms of rule quality, removing Normal and Anomaly or Rule Aggregation has the most negative impact. The former happens because when only normal rules are present, the LLM overreacts to slightly different actions such as "walking with an umbrella" compared to the rule for "walking", leading to false positives. Furthermore, without rules for anomalies as a reference, the LLM easily misses anomalies. The latter is due to that perception errors in the induction stage would lead to incorrect rules for normal. Besides, removing other strategies also decreases AUC, underscoring their significance. In summary, the proposed strategies effectively improve AnomalyRuler's performance. There is no direct positive/negative correlation between rule quantity and quality, i.e., having too few rules leads to inadequate coverage of normality and anomaly concepts while having too many rules would cause redundancy and errors.

Ablation on Hyperparameters. Fig. 3 illustrates the effects of the hyperparameters in the rule aggregation and perception smoothing modules. For rule aggregation, we conduct cross-validation on the number of batches $n = [1, 5, 10, 20]$ and the number of normal reference frames per batch $m = [1, 2, 5, 10]$. We observe that both the number of rules and AUC increase with the increases of n and m, but they start to fluctuate when $n \times m$ becomes large. For example, when $n = 20$, AUC drops from 85.9% to 72.2% as m increases because having too many reference frames (e.g., over 100) results in redundant information in a long context. For perception smoothing, we test the padding size in majority vote $p = [1, 5, 10, 20]$ and the weighting parameter in EMA $\alpha = [0.09, 0.18, 0.33, 1]$. We

Table 5: Ablation on strategies. We assess the effects of removing individual strategies in AnomalyRuler. We conduct the experiments five times with different randomly selected normal reference frames for induction and report their mean and standard deviation on the ShT dataset.

Strategy	Stage	# Rules		Accuracy		Precision		Recall		AUC	
		mean	std	mean	std	mean	std	mean	std	mean	std
w. All Below (default)	Both	42.2	4.2	81.6	1.3	90.9	0.8	63.9	2.7	84.5	1.1
w/o. Human and Environment	Both	-20.1	+1.1	-3.3	+0.8	-3.9	+0.8	-1.9	+1.6	-2.4	+2.0
w/o. Normal and Anomaly	Induction	-34.8	-1.3	-20.5	+4.3	-41.2	+7.0	-14.4	+11.6	-18.8	+1.2
w/o. Abstract and Concrete	Induction	+2.3	+2.7	-0.6	-0.2	-0.9	-0.2	-0.3	-0.4	-0.9	+0.1
w/o. Rule Aggregation	Induction	+8.5	+6.1	-9.6	+ 14.7	+1.1	+2.9	-10.7	+14.1	-15.8	+0.8
w/o. Perception Smoothing	Deduction	NA	NA	-1.7	-0.9	-1.9	+0.1	-3.8	-0.3	-3.3	+0.8

(a) n & m vs. # Rules   (b) n & m vs. AUC   (c) p & α vs. AUC
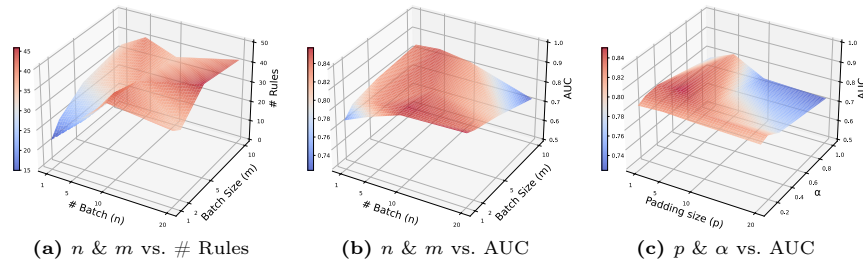
Fig. 3: Ablation on hyperparameters of the (a) (b) rule aggregation and (c) perception smoothing modules on the ShT dataset.

found $p = 5$ to be optimal for capturing the motion continuity in a video while avoiding the excessive noise that can occur with more neighborhoods. α adjusts the weight of the most recent frames compared to previous frames. A smaller α emphasizes previous frames, resulting in more smoothing but less responsiveness to recent changes. In general, increasing α from 0.09 to 0.33 improves AUC, suggesting that moderate EMA smoothing is beneficial.

6   Conclusion

In this paper, we propose AnomalyRuler, a novel rule-based reasoning framework for VAD with LLMs. With the induction and deduction stages, AnomalyRuler requires only few-normal-shot prompting without the need for expensive full-shot tuning, thereby fast steering LLMs' reasoning strengths to various specific VAD applications. To the best of our knowledge, AnomalyRuler is the first reasoning approach for one-class VAD. Extensive experiments demonstrate AnomalyRuler's state-of-the-art performance, reasoning ability, and domain adaptability. Limitations and potential negative social impact of this work are discussed in the Appendix A.1. In future research, we expect this work to advance broader one-class problems and related tasks, such as industrial anomaly detection [6, 55], open-set recognition [5, 40], and out-of-distribution detection [17, 42].

Acknowledgments

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Acsintoae, A., Florescu, A., Georgescu, M.I., Mare, T., Sumedrea, P., Ionescu, R.T., Khan, F.S., Shah, M.: Ubnormal: New benchmark for supervised open-set video anomaly detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
3. Aich, A., Peng, K.C., Roy-Chowdhury, A.K.: Cross-domain video anomaly detection without target domain adaptation. In: IEEE/CVF Winter Conference on Applications of Computer Vision (2023)
4. Bacon, F.: Novum organum (1620)
5. Bendale, A., Boult, T.E.: Towards open set deep networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2016)
6. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)
7. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: Conference on Neural Information Processing Systems (2020)
8. Cao, Y., Xu, X., Sun, C., Huang, X., Shen, W.: Towards generic anomaly detection and understanding: Large-scale visual-linguistic model (gpt-4v) takes the lead. arXiv preprint arXiv:2311.02782 (2023)
9. Chang, Y., Tu, Z., Xie, W., Yuan, J.: Clustering driven deep autoencoder for video anomaly detection. In: European Conference on Computer Vision (2020)
10. Cohen, J., Rosenfeld, E., Kolter, Z.: Certified adversarial robustness via randomized smoothing. In: International Conference on Machine Learning (2019)
11. Diao, S., Wang, P., Lin, Y., Zhang, T.: Active prompting with chain-of-thought for large language models. arXiv preprint arXiv:2302.12246 (2023)
12. Elhafsi, A., Sinha, R., Agia, C., Schmerling, E., Nesnas, I.A., Pavone, M.: Semantic anomaly detection with large language models. In: Autonomous Robots (2023)
13. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning (2015)
14. Georgescu, M.I., Ionescu, R.T., Khan, F.S., Popescu, M., Shah, M.: A background-agnostic framework with adversarial training for abnormal event detection in video. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
15. Gu, Z., Zhu, B., Zhu, G., Chen, Y., Tang, M., Wang, J.: Anomalygpt: Detecting industrial anomalies using large vision-language models. In: AAAI Conference on Artificial Intelligence (2024)

16. Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A.K., Davis, L.S.: Learning temporal regularity in video sequences. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2016)
17. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: International Conference on Learning Representations (2017)
18. Hirschorn, O., Avidan, S.: Normalizing flows for human pose anomaly detection. In: IEEE/CVF International Conference on Computer Vision (2023)
19. Jiang, A.Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D.S., Casas, D.d.l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al.: Mistral 7b. arXiv preprint arXiv:2310.06825 (2023)
20. Lee, S., Kim, G.: Recursion of thought: A divide-and-conquer approach to multi-context reasoning with language models. In: Annual Meeting of the Association for Computational Linguistics (2023)
21. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International Conference on Machine Learning (2023)
22. Li, W., Mahadevan, V., Vasconcelos, N.: Anomaly detection and localization in crowded scenes. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2013)
23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: Conference on Neural Information Processing Systems (2023)
24. Liu, W., Luo, W., Lian, D., Gao, S.: Future frame prediction for anomaly detection–a new baseline. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
25. Liu, Z., Nie, Y., Long, C., Zhang, Q., Li, G.: A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction. In: IEEE/CVF International Conference on Computer Vision (2021)
26. Lo, S.Y., Oza, P., Chennupati, S., Galindo, A., Patel, V.M.: Spatio-temporal pixel-level contrastive learning-based source-free domain adaptation for video semantic segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
27. Lo, S.Y., Oza, P., Patel, V.M.: Adversarially robust one-class novelty detection. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
28. Lu, C., Shi, J., Jia, J.: Abnormal event detection at 150 fps in matlab. In: IEEE/CVF International Conference on Computer Vision (2013)
29. Lu, Y., Yu, F., Reddy, M.K.K., Wang, Y.: Few-shot scene-adaptive anomaly detection. In: European Conference on Computer Vision (2020)
30. Lv, H., Chen, C., Cui, Z., Xu, C., Li, Y., Yang, J.: Learning normal dynamics in videos with meta prototype network. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)
31. Lv, H., Sun, Q.: Video anomaly detection and explanation via large language models. arXiv preprint arXiv:2401.05702 (2024)
32. Mao, C., Teotia, R., Sundar, A., Menon, S., Yang, J., Wang, X., Vondrick, C.: Doubly right object recognition: A why prompt for visual rationales. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
33. Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., et al.: Simple open-vocabulary object detection. In: European Conference on Computer Vision (2022)

34. Mittal, H., Agarwal, N., Lo, S.Y., Lee, K.: Can't make an omelette without breaking some eggs: Plausible action anticipation using large video-language models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
35. Morais, R., Le, V., Tran, T., Saha, B., Mansour, M., Venkatesh, S.: Learning regularity in skeleton trajectories for anomaly detection in videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2019)
36. Park, H., Noh, J., Ham, B.: Learning memory-guided normality for anomaly detection. In: IEEE/CVF Conference Computer Vision and Pattern Recognition (2020)
37. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Conference on Neural Information Processing Systems (2019)
38. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training. OpenAI Blog (2018)
39. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI Blog (2019)
40. Safaei, B., Vibashan, V., de Melo, C.M., Hu, S., Patel, V.M.: Open-set automatic target recognition. In: IEEE International Conference on Acoustics, Speech and Signal Processing (2023)
41. Seel, N.M.: Encyclopedia of the sciences of learning (2011)
42. Sharifi, S., Entesari, T., Safaei, B., Patel, V.M., Fazlyab, M.: Gradient-regularized out-of-distribution detection. In: European Conference on Computer Vision (2024)
43. Shi, C., Sun, C., Wu, Y., Jia, Y.: Video anomaly detection via sequentially learning multiple pretext tasks. In: IEEE/CVF International Conference on Computer Vision (2023)
44. Su, Y., Lan, T., Li, H., Xu, J., Wang, Y., Cai, D.: Pandagpt: One model to instruction-follow them all. arXiv preprint arXiv:2305.16355 (2023)
45. Sun, S., Gong, X.: Hierarchical semantic contrast for scene-aware video anomaly detection. In: IEEE/CVF Computer Vision and Pattern Recognition Conference (2023)
46. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
47. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
48. Tsai, Y.H., Hung, W.C., Schulter, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
49. Wang, G., Wang, Y., Qin, J., Zhang, D., Bao, X., Huang, D.: Video anomaly detection by solving decoupled spatio-temporal jigsaw puzzles. In: European Conference on Computer Vision (2022)
50. Wang, H., Zhang, X., Yang, S., Zhang, W.: Video anomaly detection by the duality of normality-granted optical flow. arXiv preprint arXiv:2105.04302 (2021)
51. Wang, W., Lv, Q., Yu, W., Hong, W., Qi, J., Wang, Y., Ji, J., Yang, Z., Zhao, L., Song, X., et al.: Cogvlm: Visual expert for pretrained language models. arXiv preprint arXiv:2311.03079 (2023)
52. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. In: Conference on Neural Information Processing Systems (2022)

53. Wu, J.C., Hsieh, H.Y., Chen, D.J., Fuh, C.S., Liu, T.L.: Self-supervised sparse representation for video anomaly detection. In: European Conference on Computer Vision (2022)
54. Yan, C., Zhang, S., Liu, Y., Pang, G., Wang, W.: Feature prediction diffusion model for video anomaly detection. In: IEEE/CVF International Conference on Computer Vision (2023)
55. You, Z., Cui, L., Shen, Y., Yang, K., Lu, X., Zheng, Y., Le, X.: A unified model for multi-class anomaly detection. Conference on Neural Information Processing Systems (2022)
56. Zaheer, M.Z., Mahmood, A., Khan, M.H., Segu, M., Yu, F., Lee, S.I.: Generative cooperative learning for unsupervised video anomaly detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
57. Zhang, H., Li, X., Bing, L.: Video-llama: An instruction-tuned audio-visual language model for video understanding. In: Conference on Empirical Methods in Natural Language Processing (2023)
58. Zhang, Y., Huang, X., Ma, J., Li, Z., Luo, Z., Xie, Y., Qin, Y., Luo, T., Li, Y., Liu, S., et al.: Recognize anything: A strong image tagging model. arXiv preprint arXiv:2306.03514 (2023)
59. Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., et al.: Least-to-most prompting enables complex reasoning in large language models. In: International Conference on Learning Representations (2023)
60. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. In: International Conference on Learning Representations (2024)
61. Zhu, Z., Xue, Y., Chen, X., Zhou, D., Tang, J., Schuurmans, D., Dai, H.: Large language models can learn rules. arXiv preprint arXiv:2310.07064 (2023)