IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 April 2024, accepted 20 April 2024, date of publication 2 May 2024, date of current version 10 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3396219

RESEARCH ARTICLE

Caption-Guided Interpretable Video Anomaly Detection Based on Memory Similarity

YUZHI SHI[1], TAKAYOSHI YAMASHITA[1], (Member, IEEE), TSUBASA HIRAKAWA[1], (Member, IEEE), HIRONOBU FUJIYOSHI[1], (Member, IEEE), MITSURU NAKAZAWA[2], YEONGNAM CHAE[2], AND BJÖRN STENGER[2]

[1]Chubu University, Kasugai, Aichi 487-0027, Japan
[2]Rakuten Institute of Technology, Rakuten Group Inc., Setagaya City, Tokyo 158-0094, Japan

Corresponding author: Yuzhi Shi (shi@mprg.cs.chubu.ac.jp)

ABSTRACT Most video anomaly detection approaches are based on non-semantic features, which are not interpretable, and prevent the identification of anomaly causes. Therefore, we propose a caption-guided interpretable video anomaly detection framework that explains the prediction results based on video captions (semantic). It utilizes non-semantic features to fit the dataset and semantic features to provide common sense and interpretability to the model. It automatically stores representative anomaly prototypes and uses them to guide the model based on similarity with these prototypes. Specifically, we use video memory to represent the content of videos, which includes video features (non-semantic) and caption information (semantic). The proposed method generates and updates a memory space during training, and predicts anomaly scores based on the memory similarities between the input video and the stored memories. The stored captions can be used as descriptions of representative anomaly actions. The proposed module can be easily integrated with existing methods. The interpretability and reliable detection performance of the proposed method are evaluated through extensive experiments on public benchmark datasets.

INDEX TERMS Caption-guidance, sentence similarity, video anomaly detection.

I. INTRODUCTION

The large number of monitoring videos has made video anomaly detection an increasingly daunting task for human operators. Consequently, video anomaly detection has become more crucial than ever before. Furthermore, depending on how anomalies are defined, anomaly detection techniques can be applied to various video understanding tasks, such as action detection, action recognition, and video classification. Given its significance, video anomaly detection has been extensively researched for decades. However, developing a video anomaly detection model is challenging, as the definition of an anomaly is subjective and depends on the specific application scenario. For instance, ''fighting'' is considered an abnormal behavior in our daily lives, yet it is a normal action in boxing matches. Additionally, the predictions of models lack interpretability.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenhua Guo.

Most previous approaches [1], [2], [3], [4] extract visual features from videos based solely on pixel changes across frames, without understanding the video content. This results in unexplainable prediction results and limits their practical application.

To address these problems, it is necessary to understand the semantic content of the videos. Semantic features are similar to human understanding. People understand video content based on the information of objects in the image and the interactions between them. Such information is typically included in video captions. Furthermore, captions are more easily understood than other explanation methods [6]. For example, visual information based approaches, such as Gradient-weighted Class Activation Mapping (Grad-CAM [6]) offers limited visual interpretability as it cannot provide clear boundaries between objects of interest and the background.

As shown in Figure 1, captions include important information needed for anomaly detection, and we can easily

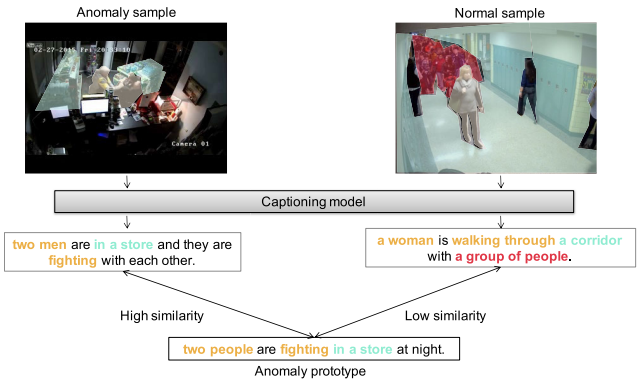


FIGURE 1. Semantic similarity. The similarity based on semantic information provided in captions is closer to the understanding of humans because it contains high-level information, such as objects and their interactions. Images from [5].

identify the positions of related objects from the image and caption. Additionally, semantic features tend to be more stable than video features because they are less affected by object appearance or capture conditions. The caption embeddings generated by a pre-trained language model also contain common sense knowledge. For example, the semantic similarity between "fighting and "violence" is greater than that between "walking" and "violence". Therefore, we utilize video captions and caption embeddings as semantic features to identify abnormal situations based on the similarity of *video memories*, where each memory contains a video feature, a video caption, and a caption embedding. Representative anomaly video memories are stored as the definitions of abnormal situations, guiding the anomaly detection model and explaining its predictions. We use this video memory to represent the content of a video and leverage the similarity among memories to guide the model and predict anomaly scores.

Our contributions in this paper can be summarized as follows:

- To address the lack of interpretability in anomaly detection models that rely on non-semantic features, we introduce video memory to represent video content and propose a novel caption-guided interpretable framework for video anomaly detection, which utilizes text as semantic features to guide the model and explain predictions.
- We visualize the anomaly actions stored in the memory space to understand what constitutes an anomaly for the models. To analyze the utility of the proposed method and demonstrate the necessity of video captions, we conduct extensive experiments.
- Our method achieves state-of-the-art performance on the ShanghaiTech [7] dataset and shows the interpretability and efficiency of the proposed approach using the UCF-Crime [5] dataset.

To the best of our knowledge, the proposed method is the first text-guided interpretable video anomaly detection model.

The rest of the paper is structured as follows: Section II introduces related work, section III outlines the proposed method, section IV summarizes the results and discusses them, and section V concludes the paper.

II. RELATED WORK

A. VIDEO ANOMALY DETECTION

Video anomaly detection has been extensively researched for decades due to its importance in security applications. Early approaches detect anomalous actions by using hand-crafted motion features, such as the histogram of oriented gradients (HOGs) [8], [9], [10], hidden Markov models (HMMs) [11], [12], sparse coding [13], and appearance features [14]. Recent approaches are predominantly based on deep learning algorithms, which utilize video features extracted by pre-trained models. At a high level, video anomaly detection approaches can be categorized into distance-based [15], [16], probabilistic [17], and reconstruction-based approaches [18], [19]. Distance-based approaches involve using the training data to create a model of "normality" and measuring deviations from this model to determine anomaly scores. Probabilistic approaches compute distances under a model in some probability space. These methods typically aim to incorporate modeling into a probabilistic framework, such as probabilistic graphical models (PGMs) or high-dimensional mixtures of probability distributions. Reconstruction approaches aim to represent the input (images or video snippets) using a high-level or compact representation learned from normal video, and then reconstruct the input using only this representation. However, these approaches are not able to explain the prediction results, making it challenging for them to be applied in real-world scenarios. We therefore propose an interpretable anomaly detection module that uses captions to guide models and interpret the definitions of anomalies.

B. INTERPRETABLE MODEL

Interpretability is crucial for deep learning models. Some methods use CAM [6] or attention maps [20] to locate important regions in an image, words in a sentence, or snippets in a video. However, their interpretation is not always intuitive. An anomaly detection model [21] leverages predictions from an action recognition model to explain its predictions. A recent method [22] utilizes object detection, tracking, and pose recognition to understand videos and employs scene graphs to explain the prediction results. However, they ignore the importance of context information in videos, *i.e.* where and when events occur, which can influence the definition of anomalous actions. For example, "two persons fighting in a boxing ring" is a normal event, whereas "two persons fighting in a kitchen" is an anomalous action. [23] introduces semantic features into the anomaly detection domain. Unlike their approach, our proposed method uses a caption-guiding module to guide the model and interpret predictions. By utilizing the video memories stored in the memory space, we can understand the definition

of anomalous situations for the model and guide the model by modifying the stored memories.

C. VISION AND LANGUAGE MODELS

Recent advancements in foundational vision and language models have led to remarkable progress in vision and language tasks. These models have also gained a degree of common sense understanding through exposure to large-scale text and image datasets. CLIP [24] connects text and images, learning visual concepts through natural language supervision. The Unified-IO [25] model offers an impressive breadth of capabilities, performing a wide variety of tasks that encompass classical computer vision, image synthesis, vision-and-language, and natural language processing. Flamingo [26] is proficient in multimodal tasks including captioning, visual dialogue, classification, and visual question answering. SwinBERT [27] is an end-to-end transformer-based architecture for video captioning and uses an adaptive learning mechanism to predict sparse attention masks. Inspired by these models, we propose a caption-guided framework for video anomaly detection. In contrast to video features extracted by pre-trained models, video captions can contextualize videos and offer insights for abnormal action detection, all in a manner that is easily comprehensible. We therefore employ video captions to explain prediction results and define the concept of an anomaly.

III. METHOD

Interpretability is necessary for anomaly detection applications in the real world because people cannot trust the models without understanding the identified anomalies. Previous works have used manual-crafted features or high-level features from pre-trained feature extractors to represent video context, but their predictions cannot be directly interpreted. While [22] uses object detection and action recognition models to understand videos, the location and temporal information in the video is not used. To address these limitations, we propose a caption-guiding module, which uses video captions to guide the model and interpret prediction results, as captions contain the necessary information for generally representing video content.

A. OVERVIEW

As shown in Figure 2, we employ multiple instance learning (MIL) to tackle the weakly-supervised video anomaly detection task, as the public datasets only contain video-level annotations. We first split an anomaly video and a normal video into 32 video snippets, respectively. Subsequently, we utilize a frozen memory generator to extract non-semantic and semantic features from the video snippets. Specifically, we use the pre-trained I3D [29] model to extract raw video features of length 2048. Additionally, we employ the pre-trained SwinBERT [27] model to generate video captions and the pre-trained MPNet [30] to extract caption embeddings from these video captions. We choose MPNet for extracting caption embeddings because it is a widely-used model for calculating sentence similarity. A caption embedding is a vector of length 786. We use the base model to project raw video features into vectors of length 32 as the optimized video features. This base model is a simple composition of three fully-connected layers. Note that the base model could be replaced by any existing model capable of representing video content through feature vectors.

We define a video memory $\mathbf{m}$ as the representation of a video snippet, which consists of an optimized video feature vector $\mathbf{f}$, a video caption $\mathbf{c}$, and a caption embedding $\mathbf{e}$. Feature $\mathbf{f}$ is a non-semantic feature, while $\mathbf{c}$ and $\mathbf{e}$ are semantic features. Video memories are fed into the caption-guiding module to generate the memory space, which stores important video memories related to anomaly actions. The stored memories are used to predict anomaly scores based on memory similarities, guide the model, and explain the predictions. We add a sigmoid layer after the caption-guiding module, a mean pooling layer, and another sigmoid layer after the base model. Finally, we employ a residual structure to reuse the optimized video features. The model outputs the snippet-level anomaly scores by combining the outputs from these two sigmoid layers. With the help of the stored video memories, the proposed method can locate anomalies based on semantic and non-semantic features, and provide interpretable predictions.

B. CAPTION-GUIDING MODULE

To enable text-guided model interpretability, we introduce a module that can utilize text as part of the video representation and guide the model based on this representation, allowing the text to generate interpretable predictions. We calculate anomaly scores based on similarities with the stored memories. Since similar memories provide redundant information, it is important to store only representative memories in the memory space to ensure each memory represents a distinct anomaly situation. On the other hand, as the parameters of the base model change during training, the video memory will also evolve. Therefore, the old memories should be removed to optimize the memory space. The caption-guiding module has three key functions: prediction of anomaly scores, generation of the memory space, and optimization of the memory space.

1) PREDICTION OF ANOMALY SCORES

The anomaly score AS is calculated based on the memory similarities between the input memory $\mathbf{m}^{input}$ and the stored memories $\{\mathbf{m}_i \mid \mathbf{m}_i \subset M\}$ in memory space M. $\mathbf{m}^{input}$ consists of a video feature $\mathbf{f}^{input}$, a video caption $\mathbf{c}^{input}$, and a caption embedding $\mathbf{e}^{input}$. To accurately characterize the relationship of video content, the memory similarity contains the non-semantic similarity based on the video features and the semantic similarity based on the caption embeddings. The calculation of non-semantic similarity s^f is presented as

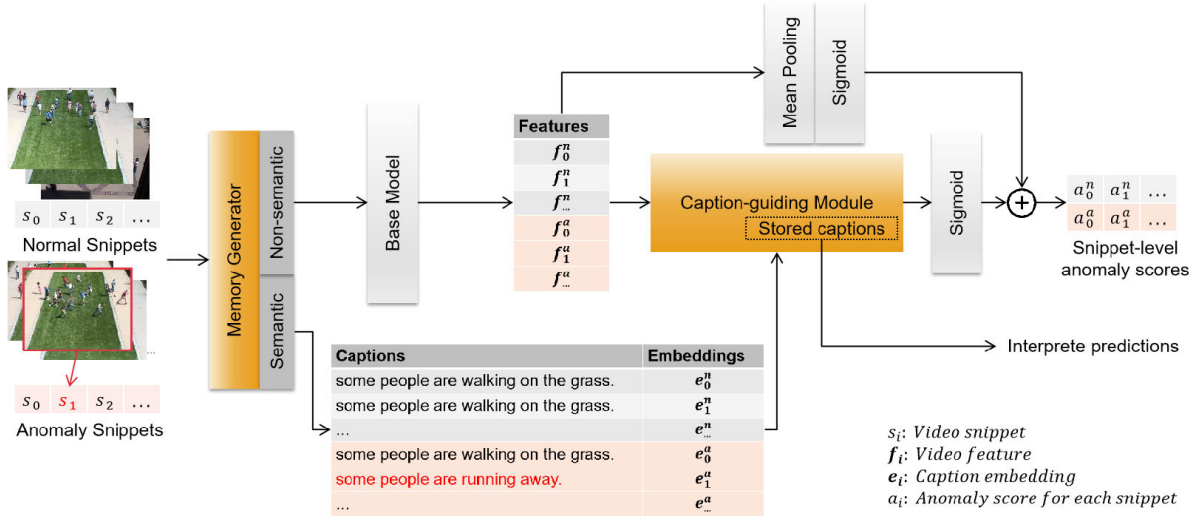


FIGURE 2. The overview of the proposed method. The proposed method contains two main modules, a memory generator and a caption-guiding module. The memory generator extracts semantic and non-semantic features as the video memory to represent video content. The caption-guiding module stores anomaly video memories to guide the model and interpret predictions using video captions. The left snippet images are cited from [28].

follows:

$$s_i^f = \mathbf{f}^{input} \cdot \mathbf{f}_i \,, \tag{1}$$

$$s^f = \text{mean}\left(\underset{\text{max}}{topK}(s_0^f, s_1^f, s_i^f, \ldots, s_I^f)\right), \quad i \subset [0, I] \,, \tag{2}$$

where I is the number of memories stored in the memory space, K is a hyperparameter, and s_i^f is the non-semantic similarity between $\mathbf{f}^{input}$ and $\mathbf{f}_i$. The number of memories I stored in the memory space changes during training.

To reduce the influence of outliers, we use the mean of the $topK$ similarities instead of the maximum or mean of all memories to represent the non-semantic similarity s^f. Semantic similarity s^e based on caption embeddings is calculated as follows:

$$s_i^e = \frac{\mathbf{e}^{input} \cdot \mathbf{e}_i}{\|\mathbf{e}^{input}\| \|\mathbf{e}_i\|} \,, \tag{3}$$

$$s^e = \text{mean}\left(\underset{\text{max}}{topK}(s_0^e, s_1^e, s_i^e, \ldots, s_I^e)\right), \quad i \subset [0, I], \tag{4}$$

where s_i^e represents the semantic similarity between $\mathbf{e}^{input}$ and $\mathbf{e}^i$. We use semantic features to calculate anomaly scores because the similarity of captions is more akin to human understanding than video features, and video captions can be directly interpreted. Additionally, the common sense included in the pre-trained language model can guide the model using the caption embeddings.

The anomaly score AS is calculated based on the non-semantic and semantic similarities:

$$AS = \frac{s^f + \theta s^e}{1 + \theta} \,, \tag{5}$$

where θ is a temperature parameter that adjusts the weight of the semantic similarity. Note that if the memory space is empty, the anomaly score is set to 0.5, as shown in Figure 3.

The anomaly score AS also represents the memory similarity between the input memory $\mathbf{m}^{input}$ and the stored memories $\mathbf{m}_i$. When the input memory is similar to the stored memories, the content of the input video is similar to the representative anomaly situation, and consequently, the anomaly score AS would be large. The anomaly score AS is calculated based on both the semantic and non-semantic features. The non-semantic features can fit the training samples, while the semantic features can provide video understanding that is closer to common sense.

2) GENERATION OF MEMORY SPACE

The memory space is the core of this module. It stores anomaly memories and outputs anomaly scores based on the similarities to the stored memories. It is generated for two purposes: explaining the prediction results and guiding the model to detect anomalies. To make full use of memory space, each memory needs to represent a different anomaly situation, and the memory space needs to store representative and distinct memories.

There are two steps for adding a new memory into the memory space. The first step is locating anomaly memory candidates $\mathbf{m}_{\mathbf{anomaly}}$ from anomaly videos. Anomaly videos contain both anomaly snippets and normal snippets. If the memories of normal snippets are put into the memory space, the model would mistakenly classify normal actions as anomalies. Therefore, normal snippets from the anomaly videos should be filtered out.

We locate anomaly memory candidates by selecting the snippets from anomaly videos that have low similarities to the snippets from normal videos. Specifically, we calculate the memory similarities between the snippets of an anomaly video and the snippets of a normal video, and take the

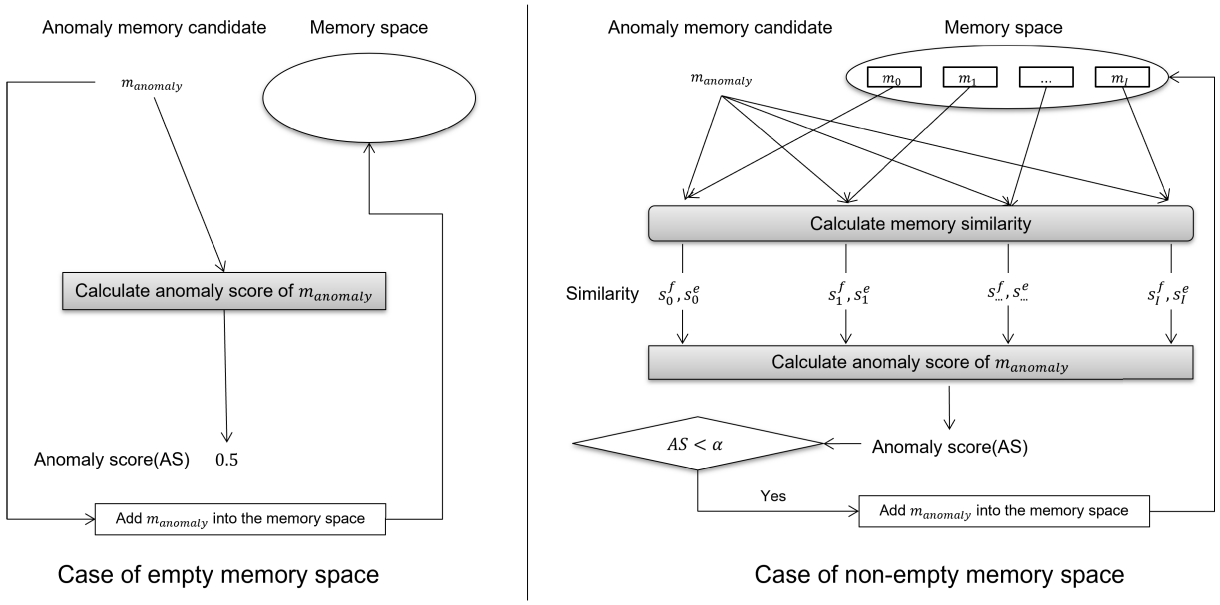


FIGURE 3. Prediction of anomaly scores and generation of memory space. The left sub-figure shows when memory space is empty, the anomaly score (AS) of the anomaly video candidate is 0.5, and it would be added to the memory space. The right sub-figure shows that when the memory space is not empty, the anomaly score is calculated based on the memory similarity with all stored memories. If the anomaly score is larger than a threshold α, the anomaly memory candidate will be added to the memory space.

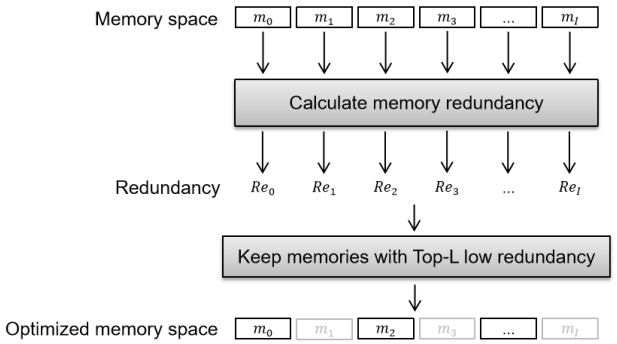


FIGURE 4. Optimization of memory space. To optimize the memory space, we calculate the redundancy of each memory and only keep the L memories with the lowest redundancy in the memory space.

mean of the memory similarities with the snippets of a normal video as the normal score for each snippet from an anomaly video. We select the snippet with the lowest normal score as an anomaly snippet candidate from each anomaly video.

The second step is to decide whether each anomaly memory candidate needs to be added to the memory space. To store representative and distinct memories, only the memory candidates with low similarities to the stored memories should be added to the memory space. To find such memories, we calculate the memory similarities between an anomaly memory candidate and all stored memories $\{\mathbf{m}_i \mid \mathbf{m}_i \subset M\}$ in the memory space. This process is the same as the calculation of the anomaly score AS, so both calculations are completed concurrently, as depicted in Figure 3.

If the AS of an anomaly memory candidate is smaller than a threshold value α, then we consider the input memory to represent a new type of anomaly situation and add it to the memory space. α is initially set to 1 and is updated through the optimization of the memory space. In this manner, the memory space is generated automatically and filled with representative anomaly memories. Therefore, the memory space can guide the model in finding anomaly actions based on the memory similarities between the stored memories.

3) OPTIMIZATION OF MEMORY SPACE

Due to the parameters of the base model changing during training, the video features $\mathbf{f}$ and video memories $\mathbf{m}$ would also change. Consequently, the meaning of old memories may become outdated, and some stored redundant memories could misguide the model. To detect anomalies efficiently, distinct and representative anomaly memories need to be kept, while redundant memories need to be removed. The optimization of the memory space is shown in Figure 4. To find redundant memories, we calculate the redundancy Re_i of the i^{th} memory as follows:

$$Re_i = \text{mean}\left(\{s^m{}_{ij} \mid s^m{}_{i0}, s^m{}_{i1}, \ldots, s^m{}_{iI}\}\right); \quad j \neq i, \tag{6}$$

$$s^m{}_{ij} = \mathbf{f}_i \cdot \mathbf{f}_j + \theta \frac{\mathbf{e}_i \cdot \mathbf{e}_j}{\|\mathbf{e}_i\| \|\mathbf{e}_j\|}, \tag{7}$$

where $s^m{}_{ij}$ denotes the similarity between $\mathbf{m}_i$ and $\mathbf{m}_j$. i and j both represent memory numbers, with the same range of $[0 \sim I]$.

We consider the memory m_i with a higher Re_i to contain less useful information. We keep the L memories with the $topL$ minimum redundancy and remove the others to optimize

TABLE 1. Comparison of frame-level AUC performance for video anomaly detection on the ShanghaiTech dataset. The proposed method uses S3R as the base model.

Method	Feature	AUC (%)
GCN-Anomaly [1]	C3D [33]	76.44
GCN-Anomaly	TSN [34]	84.44
MIST [2]	C3D	93.13
MIST	I3D	94.83
RTFM [3]	C3D	91.51
RTFM	I3D	97.21
MSL [35]	C3D	94.81
MSL	I3D	96.08
S3R [4]	I3D	97.48
Ours (S3R)	I3D + Caption	**97.69**

the memory space. Because the base model would update its parameters to fit the training dataset, the similarity of video features would change. Therefore, the threshold α needs to be updated to adapt to these changes and suppress the addition of similar memories into the memory space. We update the threshold α as follows:

$$\alpha = \max\left(\min_{topL}(Re_0, Re_1, Re_2, \ldots, Re_I)\right), \quad (8)$$

where L is a hyperparameter to limit the number of memories retained. If the number of memories in the original memory space is larger than L, only L memories are stored.

C. IMPLEMENTATION DETAILS

Since most anomaly detection datasets only have video-level annotations, we use multiple instance learning (MIL) to train models, following previous work [3], [4], [5]. For a fair comparison, we adopt the pre-trained I3D model [29] on Kinetics-400 [31] for video feature extraction. On the ShanghaiTech dataset, we train our model using the Adam [32] optimizer with a learning rate of 10^{-3}, following the training procedure of S3R [4]. On the UCF-Crime dataset [5], we train our model using the AdaGrad optimizer with an initial learning rate of 0.1, reducing it by a factor of 10 after epochs 25 and 50, respectively. During inference, the memory space would not be updated. The anomaly score is calculated based on the similarity with the stored memories.

Regarding the hyperparameters of the proposed model, we set the top-K parameter to 5, the number of stored memories L to 7, and the temperature parameter θ to 1. Additionally, we conduct memory space optimization every 3 iterations. Optimization is skipped if the number of stored memories falls below 10.

IV. EXPERIMENTS

We conduct extensive experiments to evaluate the performance and interpretability of the proposed method on two datasets: ShanghaiTech [7] and UCF-Crime [5]. Both datasets are used for weakly-supervised video anomaly detection.

Datasets: The ShanghaiTech dataset [7] contains 437 videos from 13 campus surveillance scenes. In this dataset 238 videos are used for training and 199 videos for testing in the weakly-supervised setting. The UCF-Crime [5] dataset contains 1900 surveillance videos covering 13 real-world anomalous classes such as robbery, explosion, and road accident. It contains 1610 training videos and 290 test videos. Compared to ShanghaiTech, which mainly includes pedestrian activities in a university setting, the scenes in the UCF-Crime dataset are more diverse and complex.

TABLE 2. Comparison of frame-level AUC performance for video anomaly detection on the UCF-Crime dataset. MLP is an MLP-based model, that contains 4 fully connected layers. The proposed method uses an MLP model excluding the last fully connected layer as the base model.

Method	Feature	Interpretable	AUC (%)
GCN-Anomaly [1]	TSN	✗	82.12
MIST [2]	I3D	✗	82.30
MLP	I3D	✗	82.81
RTFM [3]	I3D	✗	84.30
S3R [4]	I3D	✗	**85.99**
Ours (MLP)	I3D+Caption	✓	**84.64**

Metric: For evaluating the model performance on video anomaly detection, we calculate the Area Under Curve (AUC), a conventional threshold-independent metric [4], [5].

A. VIDEO ANOMALY DETECTION

To evaluate the performance of the proposed method for anomaly detection, we use two different base models, MLP and S3R, to evaluate the proposed method on ShanghaiTech and UCF-Crime datasets, respectively.

As shown in Table 1, when using S3R [4] as the base model, our proposed method achieves an AUC score increase of 0.21%, reaching state-of-the-art performance on the ShanghaiTech dataset. We attribute this to the fact that our method can detect anomalies based on caption embeddings, allowing it to leverage semantic information to improve the existing method.

Moreover, we evaluate the interpretability and performance of the proposed method on the public UCF-Crime dataset for video anomaly detection, as presented in Table 2. Using a multi-layer perceptron (MLP) with four fully connected layers as the base model, our proposed approach achieves an AUC score improvement of 1.79% over the standalone MLP model. We do not use S3R as the base model on the UCF-Crime dataset due to the limitation of GPU resources. However, our method still reaches a comparable performance even with the MLP model as the base model, as video captions provide necessary clues for anomaly detection. The UCF-Crime dataset contains untrimmed videos, which makes it difficult to generate accurate video captions. For example, many videos contain a logo scene, and some scenes repeat several times within a video. This limitation reduces the effect of the video captions. Despite these challenges, the proposed model can still interpret the predictions using the available video captions. More analyses are provided in Section IV-B.

TABLE 3. Comparison with different memory types. To evaluate the performance of semantic features, we leverage three different features as the memory.

Memory type	AUC (%)
Caption embedding (CE)	62.40
Video feature (VF)	82.60
Ours (CE+VF)	**84.64**

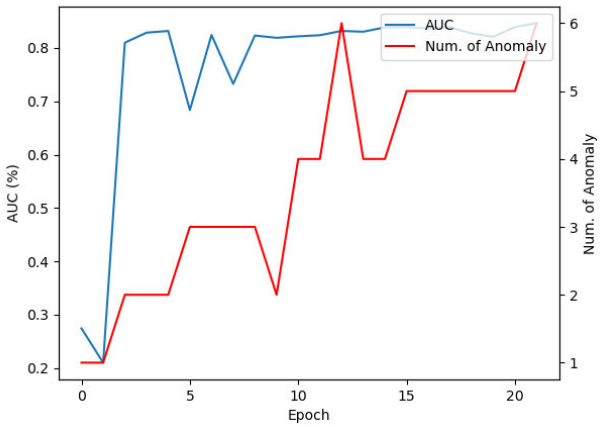


FIGURE 5. Change of memory space during training. We evaluate the influence of anomaly memories by analyzing the relationship between the number of anomaly memories in the memory space and the AUC score.

B. INTERPRETABILITY

Interpretability is a critical function for an anomaly detection model, as it requires the model not only to detect anomalous actions but also to understand the video context. The proposed model represents the video content using semantic and non-semantic features, detects anomaly actions based on memory similarities with the stored anomaly memories, and explains the predictions via video captions. We conduct several experiments to analyze the interpretability of the proposed method.

1) STRENGTH OF CAPTION EMBEDDINGS

Our method is the first to use visual captions (semantic features) for anomaly detection. Previous work extracted video features to detect anomalies. However, these features cannot be directly interpreted. In contrast, we introduce a semantic video representation by using video captions as a part of video memory to represent video content. To show the usefulness of semantic features, we compare three different memory types: video features (VF), caption embeddings (CE), and the proposed memory type (CE+VF), which contains both video features and caption embeddings.

The results are shown in Table 3. Using caption embeddings (CE) as the memory, the AUC score is 62.4%. We attribute this to the fact that the untrimmed videos in the dataset lead to some incorrect video captions, limiting the effectiveness of the caption embeddings. While CEs contain important information and common sense to guide the model, it has less information than video features (VF) and is influenced by inaccurate captions. In contrast, VFs include detailed information from the videos, allowing it to fit the training samples and achieve an AUC of 82.6%. In contrast to VF, the proposed memory type CE+VF contains common sense information that guides the model without extensive training, enabling it to understand video content and achieving the best performance. Furthermore, the proposed memory type can leverage video captions to explain the prediction results, as shown in Figure 7. Moreover, using caption embeddings as part of the video memory allows the model to output meaningful predictions.

2) MEANINGFUL PREDICTIONS

As shown in Figure 6, only the moment when a man sets fire is annotated as an anomaly. However, the fire grew larger, and the situation became more dangerous after that initial moment. The model using only video features outputs small anomaly scores for the scenes where the fire grows larger. In contrast, the model using caption embeddings predicts increased anomaly scores in the scenes where the room is full of smoke. This is because the semantic similarity between "smoke" and the descriptions of representative anomaly situations, such as "explosion" and "fire", is large. The meaningful predictions from the model using caption embeddings are more suitable for real-world applications. The definition of anomaly and the annotations in datasets are subjective. If the model is trained solely to fit the annotation data, it would lack common sense and ignore some dangerous situations. However, using video captions to guide the model allows it to output more meaningful anomaly scores.

3) CHANGE OF MEMORY SPACE DURING TRAINING

The stored memories guide the model to detect anomaly actions based on the memory similarities. For example, if there is a video memory related to "fighting" in the memory space, it helps the model to detect "fighting" actions. Therefore, we hypothesize that if the memory space stores more anomaly memories, the performance will improve. We analyze the change in the number of anomaly memories in the memory space and evaluate the AUC score during training. After each training epoch, we optimize the memory space if the number of memories is larger than 10. As a result of the optimization, only seven representative memories are kept, and we determine whether each stored memory is an anomaly memory based on the corresponding caption. As shown in Figure 5, the number of stored anomaly memories increases, and the AUC score improves during training. The stored memories guide the model via semantic similarity to reach better anomaly detection performance.

4) STORED MEMORIES

For further analysis, we show several stored captions and frames in Figure 7. The stored captions describe the anomaly actions, allowing the proposed method to recognize related anomaly actions through the memory similarities with the input video snippet. For example, the caption "two men are

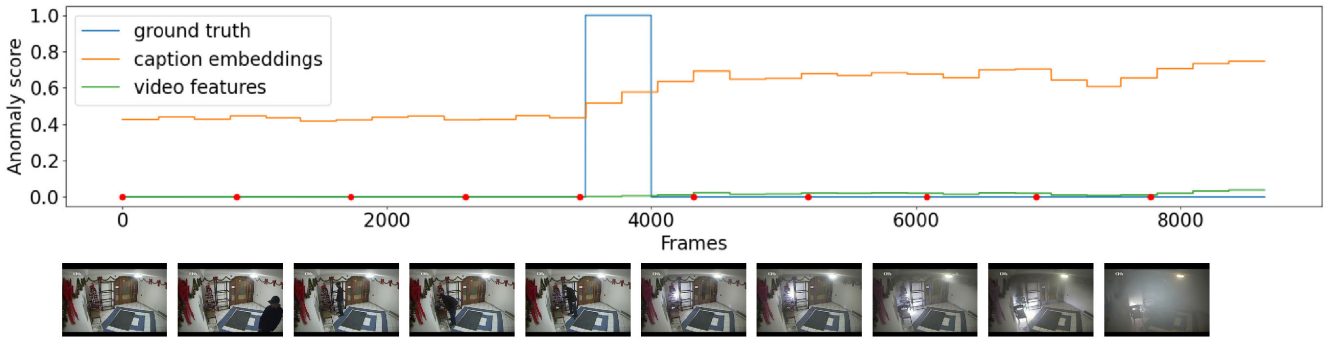


FIGURE 6. Comparison of predictions from the models with caption embeddings and with video features. The figure shows video frames at the bottom, with the timestamps of selected frames plotted as red points. The graph displays the predicted anomaly scores of two models and the annotated ground truth. The blue line represents the ground truth of anomaly. The green line represents the anomaly scores of the model using video features as the video memory. The orange line represents the anomaly scores of the proposed method that utilizes both video features and caption embeddings. The frames are taken from the UCF-Crime dataset [5].

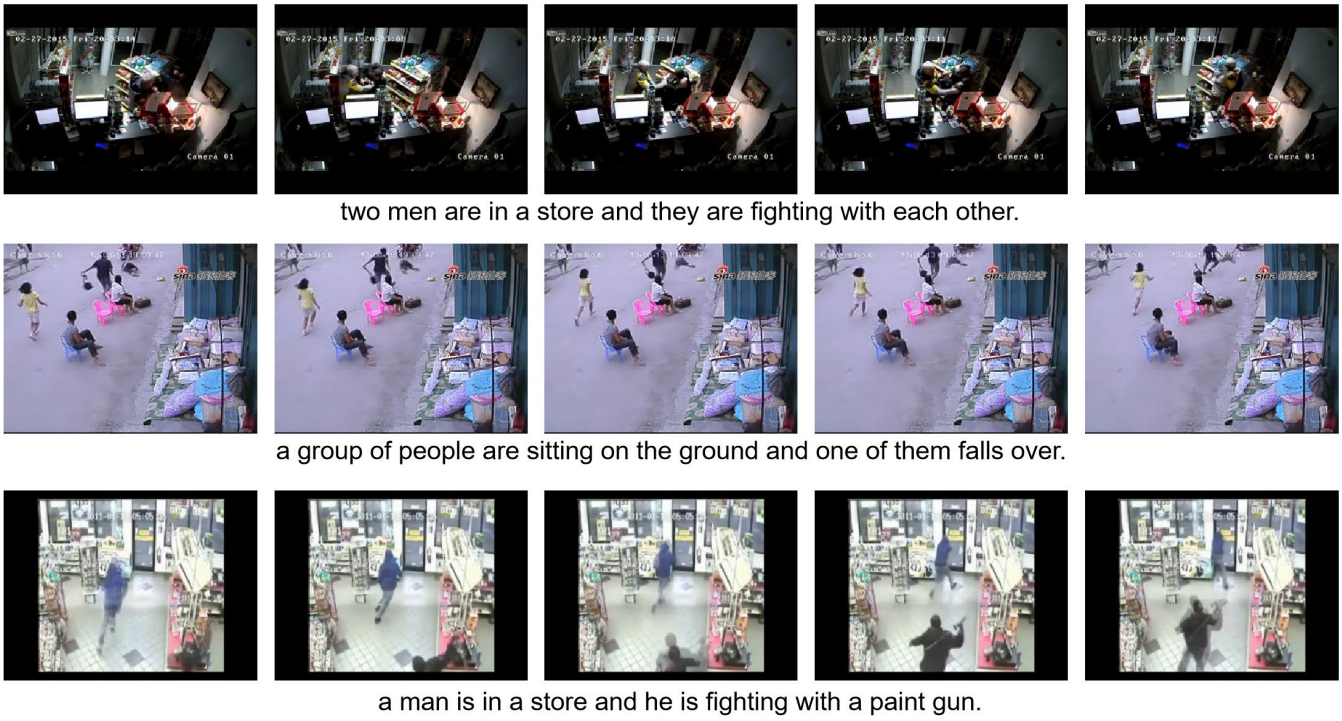


FIGURE 7. Examples of stored video memories. Five frames are sampled from each video to show the video content. The video caption is generated by the memory generator and stored in the memory space. They are the definition of anomalies for the model. The video is from the UCF-Crime dataset [5].

in a store and they are fighting with each other'' contains the word ''fighting'', which is associated with violence and has a high semantic similarity with video captions including violence-related words. By storing such memories in the memory space, the model is guided to detect anomaly actions based on these semantic similarities. Furthermore, we observe an interesting phenomenon - the memory space stores some captions that describe the scenes preceding the anomaly actions, such as ''a person is throwing a package onto a door of a house.'' We believe this can help the model detect anomalies earlier. As training progresses, the memory space becomes more stable, as the updates to the base model become less frequent and slower. Eventually the memory space holds the appropriate memories for effective anomaly detection.

C. ABLATION STUDY

To analyze the importance of memory space optimization and the influence of the proposed method on training and inference time, we conducted additional experiments.

1) MEMORY SPACE OPTIMIZATION

The optimization of the memory space removes redundant memories from the memory space and updates the threshold

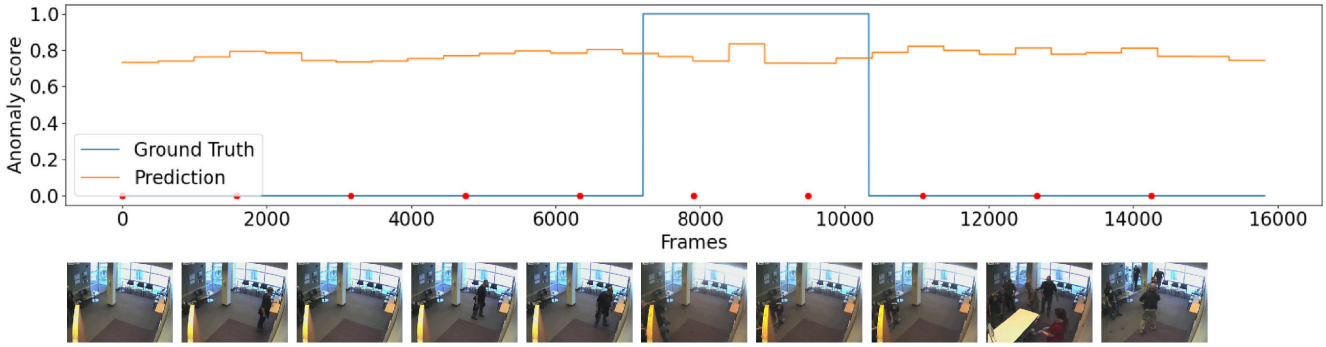


FIGURE 8. **Failure Case. The figure shows ten frame images of the video at the bottom, the timestamps of selected frames are presented in the graph using the red points. The blue line presents the ground truth of the anomaly. The orange line presents the anomaly scores of the proposed method. The video frame images are cited from [5].**

TABLE 4. **Comparison of different optimization methods. We compared three calculation methods which based on three different feature types and compare the fixed threshold α with the variable threshold. Note that we only changed the feature types used for optimization and all models use CE+VF as the video memory.**

Feature type for optimization	Threshold α	AUC (%)
Video feature (VF)	variable	83.09
Caption embedding (CE)	variable	82.55
CE+VF	0.3	83.33
CE+VF	0.4	83.56
CE+VF	0.5	83.94
CE+VF	0.6	83.54
CE+VF	variable	**84.64**

TABLE 5. **Comparison of training and inference times. We also report the number of video snippets processed per second.**

Method	Training time (h)	Inference time (snippets/s)	AUC (%)
S3R	38.33	41.67 ± 2.26	97.48
Ours (S3R)	42.67	36.29 ± 1.92	97.69

α to suppress the addition of similar memories. We compare three methods for calculating memory redundancy: based on the similarity of video features, based on the similarity of caption embeddings, and based on the similarity of video memories, respectively. Selecting redundant memories based on the video memories reaches better performance, as shown in Table 4. This suggests that leveraging video memory can correctly identify representative anomaly memories. To demonstrate the need for updating the threshold α, we compare a variable threshold with a fixed threshold. The experiments show that the variable threshold results in the best performance. This is because the base model projects the same video feature into different feature vectors to fit the training dataset during training, changing the distances among video memories. Therefore, the threshold is updated based on the stored memories to prevent adding similar video memories to the memory space.

2) TRAINING AND INFERENCE TIME

To assess the impact of our proposed method on both training and inference times, we conducted experiments comparing it with the S3R model using the ShanghaiTech dataset. Our method builds upon the S3R model as base model. We trained both models for 15,000 epochs on a NVIDIA® A100 GPU and recorded the time taken to achieve optimal performance as the training time. As shown in Table 5, due to the additional computations involved in the caption-guided memory module, our method requires more time for both training and inference compared to the S3R model. However, our approach offers interpretability through video captions and demonstrates improved performance over the base model.

D. FUTURE WORK

A failure case is depicted in Figure 8, where an arrest scene is obscured by a yellow door in the sixth and seventh frames. Our model assigns high anomaly scores to all scenes, including normal ones, due to this obscured critical event. The absence of ''arrest'' related terms in video captions further complicates anomaly detection. To improve performance, we propose exploring anomaly detection based on changes in video captions as future work.

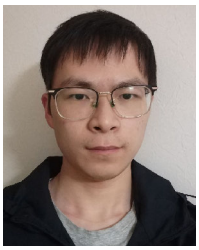
V. CONCLUSION

To improve video anomaly detection models, we propose a text-guided interpretable framework, leveraging memory similarity. By incorporating video captions, we offer interpretability to the process, guiding models in defining anomalies. Through extensive experimentation, our method has demonstrated performance gains on two public anomaly detection datasets, while also shedding light on the interpretability. Furthermore, our introduced video representation, termed ''video memory'', enables the model to produce meaningful predictions grounded in common sense, drawing from pre-trained language models.

REFERENCES

[1] J.-X. Zhong, N. Li, W. Kong, S. Liu, T. H. Li, and G. Li, ''Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1237–1246.

[2] J.-C. Feng, F.-T. Hong, and W.-S. Zheng, "MIST: Multiple instance self-training framework for video anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14004–14013.
[3] Y. Tian, G. Pang, Y. Chen, R. Singh, J. W. Verjans, and G. Carneiro, "Weakly-supervised video anomaly detection with robust temporal feature magnitude learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4955–4966.
[4] J.-C. Wu, H.-Y. Hsieh, D.-J. Chen, C.-S. Fuh, and T.-L. Liu, "Self-supervised sparse representation for video anomaly detectio," in *Proc. ECCV*, 2022, pp. 729–745.
[5] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.
[6] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," 2016, *arXiv:1610.02391*.
[7] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection—A new baseline," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6536–6545.
[8] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 18–32, Jan. 2014.
[9] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1932–1939.
[10] R. V. H. M. Colque, C. Caetano, M. T. L. de Andrade, and W. R. Schwartz, "Histograms of optical flow orientation and magnitude and entropy to detect anomalous events in videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 3, pp. 673–682, Mar. 2017.
[11] T. Hospedales, S. Gong, and T. Xiang, "A Markov clustering topic model for mining behaviour in video," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 1165–1172.
[12] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1446–1453.
[13] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *Proc. CVPR*, Jun. 2011, pp. 3313–3320.
[14] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *Proc. CVPR*, Jun. 2011, pp. 3449–3456.
[15] M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," *Comput. Vis. Image Understand.*, vol. 172, pp. 88–97, Jul. 2018.
[16] H. T. Tran and D. Hogg, "Anomaly detection using a convolutional winner-take-all autoencoder," in *Proc. BMVC*, 2017, pp. 1–12.
[17] R. Hinami, T. Mei, and S. Satoh, "Joint detection and recounting of abnormal events by learning deep generic knowledge," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3619–3627.
[18] H. Park, J. Noh, and B. Ham, "Learning memory-guided normality for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14372–14381.
[19] T. N. Nguyen and J. Meunier, "Anomaly detection in video sequence with appearance-motion correspondence," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1273–1283.
[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, 2017, pp. 5998–6008.
[21] S. Szymanowicz, J. Charles, and R. Cipolla, "Discrete neural representations for explainable anomaly detection," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 1506–1514.
[22] K. Doshi and Y. Yilmaz, "Towards interpretable video anomaly detection," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 2654–2663.
[23] W. Chen, K. T. Ma, Z. J. Yew, M. Hur, and D. A.-A. Khoo, "TEVAD: Improved video anomaly detection with captions," in *Proc. CVPRW*, Jun. 2023, pp. 5548–5558.
[24] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, 2021.
[25] J. Lu, C. Clark, R. Zellers, R. Mottaghi, and A. Kembhavi, "UNIFIED-IO: A unified model for vision, language, and multi-modal tasks," in *Proc. ICLR*, 2023, pp. 1–34.
[26] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," 2022, *arXiv:2204.14198*.
[27] K. Lin, L. Li, C.-C. Lin, F. Ahmed, Z. Gan, Z. Liu, Y. Lu, and L. Wang, "SwinBERT: End-to-end transformers with sparse attention for video captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17928–17937.
[28] Y. Hu, Y. Zhang, and L. S. Davis, "Unsupervised abnormal crowd activity detection using semiparametric scan statistic," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 767–774.
[29] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. CVPR*, Jul. 2017, pp. 4724–4733.
[30] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "MPNet: Masked and permuted pre-training for language understanding," in *Proc. NeurIPS*, vol. 33, 2020, pp. 16857–16867.
[31] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6299–6308.
[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015, pp. 1–15.
[33] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4489–4497.
[34] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. ECCV*, 2016, pp. 20–36.
[35] S. Li, F. Liu, and L. Jiao, "Self-training multi-sequence learning with transformer for weakly supervised video anomaly detection," in *Proc. AAAI*, 2022, pp. 1–9.



YUZHI SHI received the B.S. degree in software engineering from Jiaxing University, China, in 2019, and the M.S. degree from the Department of Computer Science, Chubu University, Japan, in 2021.

TAKAYOSHI YAMASHITA (Member, IEEE) received the Ph.D. degree from the Department of Computer Science, Chubu University, Japan, in 2011. He was with OMRON Corporation, from 2002 to 2014. He was a Lecturer with the Department of Computer Science, Chubu University, from 2014 to 2017, where he was an Associate Professor with the Department of Computer Science, from 2017 to 2021. He has been a Professor with the Department of Computer Science, Chubu University, since 2021. His current research interests include object detection, object tracking, human activity understanding, pattern recognition, and machine learning. He is a member of the IEICE and the IPSJ.

TSUBASA HIRAKAWA (Member, IEEE) received the Ph.D. degree from the Department of Computer Science, Hiroshima University, Japan, in 2017. From 2017 to 2019, he was a Researcher Fellow with Chubu University, Japan, where he has been a specially appointed Associate Professor with Chubu Institute for Advanced Studies, since 2019. He was a Visiting Researcher with ESIEE Paris, France, in 2014 and 2015. He was a Fellowship of Japan Society for the Promotion of Science, from 2014 to 2017. He has been a Lecturer with the Department of Computer Science, Chubu University, since 2021.

HIRONOBU FUJIYOSHI (Member, IEEE) received the Ph.D. degree in electrical engineering from Chubu University, Japan, in 1997. From 1997 to 2000, he was a Postdoctoral Fellow with the Robotics Institute of Carnegie Mellon University, Pittsburgh, PA, USA, working on the DARPA video surveillance and monitoring (VSAM) effort and the humanoid vision project for the HONDA humanoid robot. He is currently a Professor with the Department of Robotics, Chubu University. From 2005 to 2006, he was a Visiting Researcher with the Robotics Institute, Carnegie Mellon University. His research interests include computer vision, video understanding, and pattern recognition. He is a member of the IEICE and the IPSJ.

MITSURU NAKAZAWA received the Ph.D. degree in engineering from Keio University, in 2011. From 2008 to 2011, he was a JSPS Research Fellow for Young Scientists (DC1). From 2011 to 2015, he was a Postdoctoral Researcher with Osaka University. He is currently the Assistant Manager and a Principal Research Scientist with the Rakuten Institute of Technology, Rakuten Group Inc. His research interest includes visual understanding from image or video data for business process automation.

YEONGNAM CHAE received the Ph.D. degree in computer science from KAIST, in 2013. Currently, he is the Assistant Manager of the Rakuten Institute of Technology, Rakuten Group Inc. His research interests include face recognition, scene text recognition, and action recognition.

BJÖRN STENGER received the Diploma (M.Sc.) degree from the University of Bonn, Germany, in 2000, and the Ph.D. degree from the University of Cambridge, U.K., in 2004. He is currently leading the Vision Program, Rakuten Institute of Technology, Rakuten Group Inc. He was with the Toshiba R&D Center and the Toshiba Research Europe, before joining the Rakuten Institute of Technology. His current research interests include image and video understanding, image enhancement, and generative AI.