Unveiling Context-Related Anomalies: Knowledge Graph Empowered Decoupling of Scene and Action for Human-Related Video Anomaly Detection

Chenglizhao Chen[1,2] Xinyu Liu[1,2] Mengke Song[1,2†] Luming Li[1,2] Shaojiang Yuan[1,2] Xu Yu[1,2] Shanchen Pang[1,2,3]
[1]Qingdao Institute of Software, College of Computer Science and Technology,
China University of Petroleum (East China)
[2]Shandong Key Laboratory of Intelligent Oil & Gas Industrial Software
[3]State Key Laboratory of Chemical Safety

***Abstract*—Video anomaly detection methods are mainly classified into two categories based on their primary feature types: appearance-based and action-based. Appearance-based methods rely on low-level visual features like color, texture, and shape, learning patterns specific to training scenes. While effective in familiar settings, they struggle with unknown or altered scenes due to poor generalization and limited understanding of action-scene relationships. In contrast, action-based methods focus on detecting action anomalies but often overlook contextual scene associations, leading to misjudgments (e.g., running on a street being deemed normal without considering scene context). To overcome these limitations, we propose a novel decoupling-based anomaly detection architecture (DecoAD). Its core lies in the decoupling and interweaving of scenes and actions, enabling explicit modeling of their complex relationships. By reconstructing these interactions using knowledge graphs, DecoAD achieves a deeper understanding of behaviors and contexts. This design ensures strong performance in both known and unknown scenarios, significantly enhancing generalization. To evaluate its effectiveness in dynamic scenes and its ability to handle scene-related anomalies, we introduce UFSR, the first video anomaly detection dataset featuring dynamic scenes and scene-related anomalies. DecoAD supports fully-supervised, weakly-supervised, and unsupervised settings, improving AUC on UBnormal by 1.1%, 3.1%, and 2.1% in fully-supervised, weakly-supervised, and unsupervised settings, and on UFSR by 1.2% and 8.2% in weakly-supervised and unsupervised settings. The source code and datasets are available at: https://github.com/liuxy3366/DecoAD.**

***Index Terms*—Human-Related Video Anomaly Detection, Knowledge Graph, Scene-Action Interweaving, Deep Learning.**

I. Introduction

Video anomaly detection (VAD) is a crucial task in security surveillance, aiming to identify abnormal events in video sequences [1]–[5]. Among its branches, human-related VAD is significant, covering anomalies like aggressive actions, unauthorized entries, and scene-related anomalies. Scene-related anomalies, which are context-related, arise from the interaction between actions and the environment (e.g., running in a park is normal, but on a busy road is abnormal). The concept of scene-related anomalies was first explored in UAV surveillance research [6], as seen in detecting restricted parking violations. However, current VAD methods often overlook the interplay between actions and their surroundings, leading to poor detection of scene-related anomalies. To overcome this, we propose a novel approach that explicitly models the scene-action relationship, thus improving the detection of context-driven anomalies.

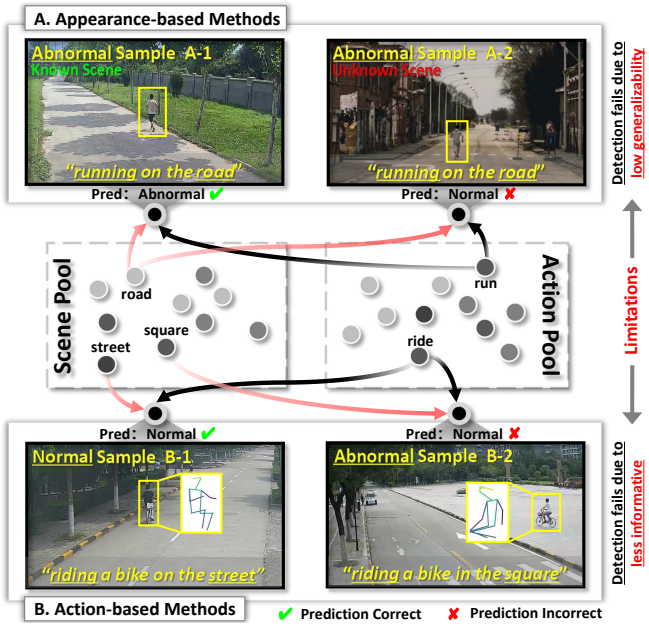

Fig. 1. Reveal the limitations of existing methods: appearance-based methods fail to detect anomalies due to their low generalizability (**A**), action-based methods fail due to their less informative (**B**). "Known Scene" refers to the scene present in the training set, and "Unknown Scene" refers to the scene not present in the training set or those that have significant changes.

To clearly compare and understand the characteristics of various video anomaly detection methods, we classify and analyze existing approaches from the perspective of feature independence, dividing them into appearance-based methods and action-based methods according to the primary feature types they rely on. Appearance-based methods typically rely on low-level visual features (e.g., color, texture, and shape) to capture human behavior [2], [4], [5]. By accurately recognizing pixel patterns in familiar scenes, these methods can effectively perform anomaly detection tasks in known environments. However, their limitation lies in their inability to deeply understand the intrinsic relationships between actions and scene context. Consequently, when significant changes occur in the scene, these methods often exhibit poor generalization performance due to their heavy reliance on low-level visual features. As illustrated in Fig. 1-A, appearance-based

† Corresponding author: Mengke Song (songsook@163.com)

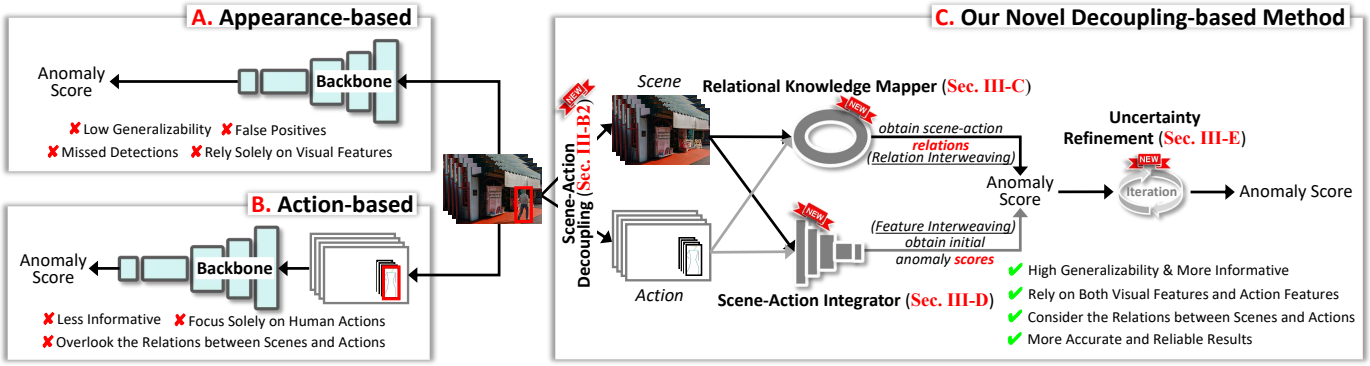

Fig. 2. Compared to appearance-based methods (**A**), which only rely on low-level visual features, and action-based methods (**B**), which ignore the relationship between scenes and human actions, our decoupling-based method (**C**) introduces the concept of "Scene-Action Interweaving". Fully considering the complex connections between actions and the surrounding environment in different video clips.

methods can successfully detect a person running on a road in a known scene but may fail in an unknown environment. In contrast, action-based methods utilize high-level features such as skeleton data and pose estimation [7], [8], significantly enhancing the model's generalization ability. These methods focus on identifying action anomalies, such as running or fighting [9]–[11], and demonstrate strong robustness across different scenarios. However, they often neglect the relationship between scene context and actions, making it challenging to accurately determine anomalies in the absence of sufficient contextual information. For example, as shown in Fig. 1-B, current methods fail to distinguish between cycling on a street and cycling in a square. This lack of scene information leads to detection failures. Therefore, relying solely on action features is insufficient to effectively handle the complex interplay between scenes and actions.

An effective video anomaly detection method should be capable of understanding the behavior in the video and clarifying the relationship between scenes and actions to accurately determine whether the behavior is abnormal. However, existing methods typically use only one type of feature (i.e., appearance features or action features) as the primary input, or simply fuse the two features without effectively modeling the relationship between appearance and action features, as shown in Fig. 2-A and B. These methods overly rely on implicit learning mechanisms to capture correlations between the data, but such implicit correlations cannot fully or accurately reflect the inherent complexity of relationships in the video, especially when facing dynamically changing scenes. As a result, the relationship modeling becomes disordered, which in turn affects the accuracy and reliability of video anomaly detection.

To effectively capture the complex relationships between scenes and actions and address scene-related anomalies at their root, we propose a novel **dec**oupling-based human-related video **a**nomaly **d**etection architecture (DecoAD). DecoAD consists of two operations: "decoupling" and "interweaving". "Decoupling" provides the foundation for independent modeling, and "interweaving" reconstructs relationships based on the independent models. The two work together to enhance the model's ability to understand complex behaviors and scenes. Simultaneously, we innovatively introduce the concept of "Scene-Action Interweaving", aiming to explicitly model the intricate relationships between scenes and actions. This method comprises two main components: "Relation Interweaving" and "Feature Interweaving". "Relation Interweaving" focuses on modeling and understanding the complex interaction relationships between scenes and actions, while "Feature Interweaving" concentrates on combining scene and action features to capture scene-related and interrelated characteristics that reflect the environment and behavior. DecoAD integrates these two components, by explicitly modeling the relationships between scenes and actions, enhancing scene-related anomaly detection, overall performance, and model generalization for robust handling of unseen scenarios.

DecoAD comprises four key components, as shown in Fig. 2-C: Scene-Action Decoupling (Sec. III-B2), Relational Knowledge Mapper (Sec. III-C), Scene-Action Integrator (Sec. III-D), and Uncertainty Refinement (Sec. III-E). First, Scene-Action Decoupling separates scenes and associated human actions from video segments. Then, Relational Knowledge Mapper applies "Relation Interweaving" to capture intricate interactions by combining related elements from different video segments. Following this, Scene-Action Integrator performs "Feature Interweaving" to generate initial anomaly scores, indicating the likelihood of anomalous behavior. Finally, Uncertainty Refinement iteratively refines these scores, ensuring more accurate anomaly detection results.

Our contributions are summarized as follows:

- We propose a novel decoupling-based video anomaly detection framework, **DecoAD**. By decoupling and intertwining scenes and actions, DecoAD effectively addresses the challenges of scene-related anomaly detection.
- We have designed four core modules: Scene-Action Decoupling (SAD), Relational Knowledge Mapper (RKM), Scene-Action Integrator (SAI), and Uncertainty Refinement (UR). These modules work together to enhance the accuracy and robustness of anomaly detection.
- We constructed a novel video anomaly detection dataset, **UFSR**, which focuses on dynamic scenes and includes scene-related anomaly data.
- DecoAD supports fully-supervised, weakly-supervised, and unsupervised settings, delivering competitive results on four datasets: NWPU Campus, UBnormal, ShanghaiTech Campus, and UFSR.

II. Related Works

A. Video Anomaly Detection

Video anomaly detection has long been a challenge in computer vision [12]–[16]. Early methods treated it as an unsupervised out-of-distribution task, using only normal samples for training [17]–[20]. However, these methods, relying on manually crafted features and statistical models, often had limited generalization and robustness. With the advancement of deep learning [21], [22], a wide array of new unsupervised learning methods have emerged in recent years [18], [23]. These methods aim to better learn normal behavior patterns. Due to the difficulty in annotating abnormal video data, unsupervised video anomaly detection has received widespread research attention. However, it is challenging to cover all normal samples during the training phase, often leading to higher false positive rates. To address this challenge, researchers have proposed weakly-supervised video anomaly detection methods [24]–[26], primarily relying on the multiple instance learning framework to compensate for the absence of video-level labels. By striking a balance between annotation costs and detection performance, weakly-supervised methods have shown considerable effectiveness. As research progresses, some datasets [27] have begun to provide frame-level annotations, opening up new possibilities for fully-supervised video anomaly detection [11], and allowing existing fully-supervised models to achieve higher detection accuracy.

In response to the diverse application demands of video data, we propose a novel video anomaly detection method that is flexible and applicable to unsupervised, weakly-supervised, and even fully-supervised learning scenarios.

B. Human-Related Video Anomaly Detection

Detecting anomalies in human-related videos is challenging due to the complexity and diversity of human behavior. Existing methods are typically divided into appearance-based [28]–[30] and action-based [7], [11], [16] approaches. Appearance-based methods rely on low-level visual features, such as color, texture, and shape [33]–[37], to learn pixel patterns for anomaly detection. While these methods perform well in familiar scenes, their accuracy significantly drops when applied to novel or drastically changing scenes. This limitation arises from their inability to capture the intricate relationships between actions and surrounding contexts, especially under varying lighting conditions or camera angles. For example, [30] proposed a feature deviation-based anomaly detection method that focuses on statistical relationships in feature space but lacks contextual scene modeling. Similarly, [28] optimized memory consumption and inference efficiency through a binarized network but remains centered on visual features, making it ineffective for capturing the interplay between actions and scenes in complex scenarios. Additionally, [29] introduced a snippet-level anomalous attention mechanism that improves anomaly localization via attention optimization but still lacks explicit modeling of scene-action relationships.

On the other hand, action-based methods identify potential anomalies by analyzing the temporal dynamics of human actions [7], [16]. These methods typically rely on high-level features, such as skeletal data or pose estimation, and exhibit strong generalization capabilities, particularly in handling diverse scenes. For instance, [7] proposed a motion prior-based regularity learning method that explicitly models the probability distribution of skeletal motion features, significantly improving the accuracy of anomaly detection in skeletal videos. Similarly, [16] employed a hierarchical spatio-temporal graph convolutional network that combines high-level and low-level graph representations to detect anomalies in individual behaviors and group interactions, demonstrating robustness across various scenarios. However, action-based methods often overlook scene context, meaning actions like running may be misclassified as normal without considering whether they occur on a beach or in the middle of a road.

Recently, researchers have explored using pretrained multimodal vision-language models or large language models (LLMs) for video anomaly detection [38], [39]. While these models show potential in handling complex relationships, they require substantial computational resources and are limited by the accuracy of text descriptions. Moreover, LLMs are not designed to process spatiotemporal visual features, which are crucial for video anomaly detection. In contrast, the DecoAD method decouples and intertwines scene and action features in a targeted manner, capturing their complex relationships, and thus is more suitable for dynamic and diverse video scenarios.

C. Knowledge Graph

Knowledge graph is a complex graph-like data structure that organizes and represents knowledge to reveal relationships and connections between data [40], [41]. It is widely applied in various fields, such as search engine optimization [42], recommendation systems [43], and social network analysis [44]. Knowledge graphs effectively integrate and correlate vast amounts of information in these applications, providing users with more accurate and insightful results. By effectively linking and organizing large amounts of information, knowledge graphs can provide users with more accurate and insightful results. Similarly, in detecting anomalies in different or new situations, knowledge graphs can help address the limitations of traditional methods that struggle to fully capture the complex relationships between actions and the environment.

Our research introduces knowledge graphs into video anomaly detection by decomposing video content into actions and background elements, using knowledge graphs to describe and understand their relationships. The core of this approach lies in the “decoupling” and “interweaving” operations. In the “decoupling” stage, unlike [45], which addresses anomaly detection by separating spatiotemporal features, we focus on decoupling scene and action information to tackle critical challenges in scene-related anomaly detection. Specifically, in the “decoupling” stage, we independently model scene and action features to ensure the model accurately captures their respective characteristics. In the “interweaving” stage, knowledge graphs are used to reconstruct the complex relationships between scene and action features, achieving semantic integration. This design fully leverages the contextual associations between scenes and actions, enhancing the model’s semantic understanding and accuracy in video anomaly detection.

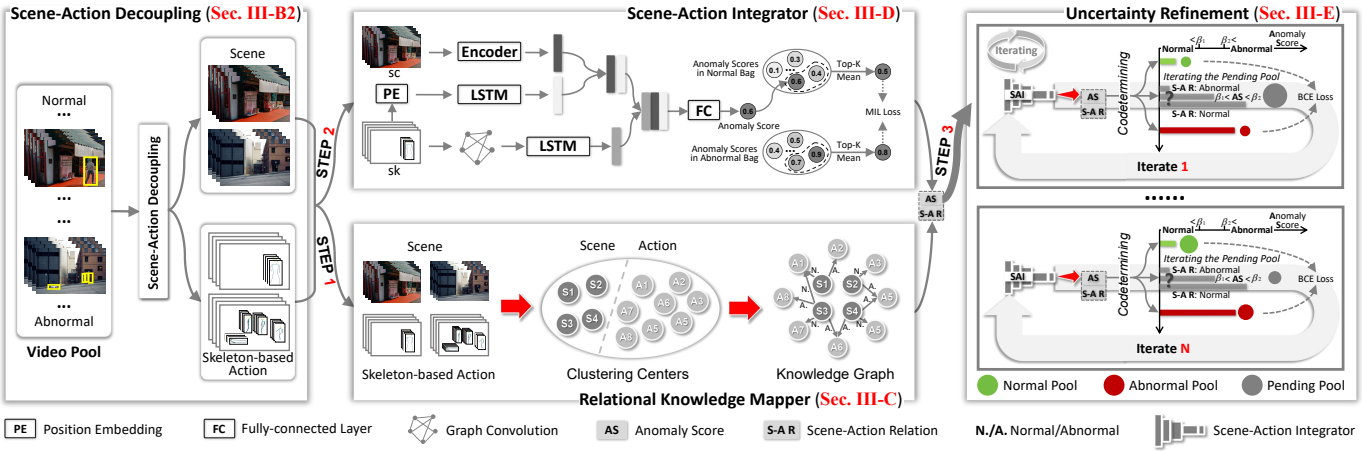

Fig. 3. Pipeline of the proposed DecoAD. DecoAD consists of three steps — Step1: Relational Knowledge Mapper (RKM), Step2: Scene-Action Integrator (SAI) and Step3: Uncertainty Refinement (UR).

III. Proposed Method

A. Method Overview

Our proposed method, DecoAD, as illustrated in Fig. 3, consists of four main components: Scene-Action Decoupling (Sec. III-B2), Relational Knowledge Mapper (Sec. III-C), Scene-Action Integrator (Sec. III-D), and Uncertainty Refinement (Sec. III-E).

In the Scene-Action Decoupling process, we first separate a video clip into scenes and their associated skeleton-based actions. Next, in Step1, through the Relational Knowledge Mapper, these actions and scenes are interwoven with actions and scenes from different video clips under the supervision of the original video samples. This involves constructing a knowledge graph to understand the relationships between scenes and actions, thereby deriving scene-action relations. In Step 2, the Scene-Action Integrator generates the initial anomaly scores, indicating the likelihood of anomalies in the video clip. Finally, in Step 3, we introduce Uncertainty Refinement to enhance the model's ability to handle ambiguous samples or borderline cases, further improving detection accuracy.

B. Preliminaries

1) Scene-Action Interweaving: Building on the existing human-related video anomaly detection methods [2], [11], it is essential to emphasize integrating scene context with human actions for more effective anomaly detection. Current approaches, whether appearance-based [3], [4] or action-based [9], [10], can recognize abnormal human actions like running or fighting. However, they frequently fail to consider the context of the scenes and actions, which can be crucial for accurately identifying scene-related anomalies.

Thus, as mentioned in Sec. I, we propose the concept of "Scene-Action Interweaving" for the first time. By decoupling scenes and human actions in video clips and interweaving them with elements from other video clips, we explore and understand the complex relationships and interactions between these scenes and actions. By combining and analyzing diverse elements from different video clips, we form a comprehensive semantic network, thereby enhancing the detection of context-related anomalies (i.e., scene-related anomalies).

2) Scene-Action Decoupling: The core of "Scene-Action Interweaving" is integrating scenes and actions with another video clip to explore their complex relationships and capture comprehensive interactions. However, in traditional methods, scenes and actions are usually treated as an overall feature for processing. This approach may lead to the interference of scene information on action features or the masking of scene details by action features, reducing the model's ability to understand complex scenes. To solve this problem, we have designed an innovative Scene-Action Decoupling mechanism in this study. Firstly, we decouple the scenes and their related actions within each video clip. For the extraction of human actions, we employ a human skeleton extraction tool, similar to the methods used in existing human-related video anomaly detection research [2], [11]. Specifically, we derive skeletal data a from the video clip V as a representation of actions[1], and simultaneously extract the positional information pos of each skeleton for subsequent operations, as shown in Fig. 4-❶:

$$\langle a,\ pos\rangle = \mathrm{SE}(\mathrm{V}), \tag{1}$$

where SE denotes the human skeleton extraction tool[2].

If action information is not removed and scene data containing actions is used directly, the action information may be considered noise, increasing the complexity of the model's processing and making the detection results unstable[3]. Additionally, since the scene data contains irrelevant action information, the model may learn unrelated features, affecting its generalization ability on new data.

To prevent action information from affecting detection results, we need to remove these elements from the scene. First, using the extracted positional information pos, we generate an action mask $mask$ with an image segmentation tool, as shown in Fig. 4-❷. Then, utilizing this mask with an image inpainting tool [47], we erase the actions from the video frames, thereby obtaining clear scene data s, as shown in Fig. 4-❸.

[1]In this study, we treat skeletal data as equivalent to actions, as actions can be effectively represented by skeletons.

[2]AlphaPose [46] is used here; any state-of-the-art human skeleton extraction tool can be applied.

[3]The performance of the model using scene data without removed action information is shown in Table II and Table III in the "Ours[3]" row.

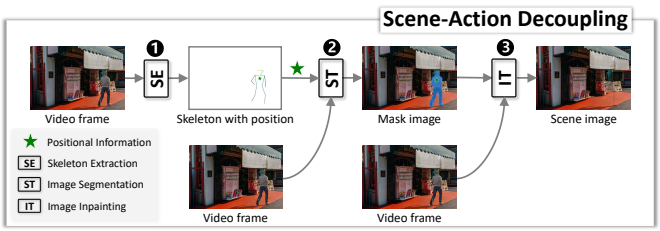

Fig. 4. Pipeline for processing image in Scene-Action Decoupling.

$$mask = \mathrm{ST}(\mathrm{V}, pos), \tag{2}$$

where ST denotes the image segmentation tool[4].

$$s = \mathrm{IT}(\mathrm{V}, mask), \tag{3}$$

where IT denotes the image inpainting tool[5].

Having successfully decoupled the video clips into scenes and associated human actions, we now proceed to examine the interrelationships between these elements.

C. Relational Knowledge Mapper

Existing methods primarily capture and represent relationships between data through implicit associations embedded within the learning mechanisms of deep learning models, rather than explicitly defining and modeling these relationships. For instance, deep learning models typically learn such associations by processing large volumes of data and labels, with these relationships encoded indirectly in the model's weights and structure. While this approach may suffice for relatively simple detection tasks, it presents significant limitations when handling more complex scenarios. Specifically, implicit learning lacks the capacity to explicitly represent contextual relationships between elements, resulting in an incomplete understanding of data interactions. This drawback is particularly pronounced when training data is sparse or when contextual dependencies (i.e., scene-action relationships) are intricate and critical for accurate anomaly detection.

To address these challenges, our proposed Relational Knowledge Mapper (RKM) explicitly models the relationships between scenes and actions using a knowledge graph. This explicit modeling ensures that the complex and context-dependent interactions between elements are clearly defined, enabling the system to go beyond the limitations of implicit learning. By leveraging the structured representation capabilities of knowledge graphs, our method captures not only direct relationships but also higher-order interactions, significantly improving the model's ability to generalize and perform reliably in diverse and dynamic environments.

As shown in Figure 3-Step1, we propose an explicit association method, the RKM, for "Relation Interweaving". This leverages the powerful representation capabilities of knowledge graphs to explicitly integrate high-level features, providing a deep understanding of the relationships between scenes and actions. This is crucial for improving the accuracy of anomaly detection. Additionally, this method has a flexible updating mechanism that can represent new relationships by adding new nodes and edges, thereby adapting to continuously changing data and environments.

In a supervised setting, given a training set, the construction of the RKM involves four processes: clustering, combining, constructing, and updating, as shown in Fig. 5. In an unsupervised setting, since only the "normal" relationship exists between scenes and actions in the training set, the combining operation is omitted.

1) Clustering: It is unrealistic to treat all data as independent information for constructing RKM. Clustering enables us to more effectively understand and categorize complex data structures. By grouping similar scenes and actions, clustering significantly enhances the manageability and accuracy of data analysis. For static scenes, where only the people move and the scene remains unchanged ($e.g.$, videos filmed with cameras at fixed angles), intuitively, when we already know the number of categories[6] for scenes and actions, we can simply put these scenes and actions in that category and find the centers without doing clustering. In contrast, dynamic scenes feature a variable number of elements in motion, including both the scenes and the people ($e.g.$, videos captured by handheld or moving cameras), require clustering (Fig. 5-❶) to unify similar scenes into the same scene category, thus simplifying scene complexity and reducing scene categories. This process groups similar scenes and actions to ensure data accurately reflects the situation, while also reducing the number of scene categories, making subsequent processing more efficient.

Given any decoupled scene and action from the dataset, we first cluster these two elements using the K-means clustering algorithm to obtain the cluster centers of the actions and scenes from normal and abnormal videos. We technically set the number of clustering centers of actions within normal and abnormal videos as θ_{fn} and θ_{fa} for each clip by the distribution statistics in the datasets[7]. For the unsupervised setting, only normal videos in the training set are clustered, with the cluster centers denoted as θ_{fn} The number of clustering centers of scenes is the same as the number of video scene categories.

By clustering actions and scenes, it not only simplifies the complexity of the data but also significantly enhances processing efficiency and classification accuracy. Moreover, it strengthens the robustness of the video analysis framework, enabling the model to perform anomaly detection more reliably when dealing with varied and complex video data.

2) Combining: Since the clips of the abnormal video may contain the content of the normal actions, we combine these normal actions clustering centers with the same normal actions clustering centers in normal videos (Fig. 5-❷). This is achieved by calculating the cosine similarity (Sim) between these cluster centers, which is denoted by:

$$\mathrm{Sim}(\mathbf{A}^{\mathrm{fn}}, \mathbf{A}^{\mathrm{fa}}) = \frac{\mathbf{A}^{\mathrm{fn}} \cdot \mathbf{A}^{\mathrm{fa}}}{\left\|\mathbf{A}^{\mathrm{fn}}\right\|_2 \cdot \left\|\mathbf{A}^{\mathrm{fa}}\right\|_2}, \tag{4}$$

[4]Segment Anything Model (SAM) [48] is used here; any state-of-the-art image segmentation tool can be applied.

[5]Inpainting Anything Model (IAM) [49] is used here; any state-of-the-art image inpainting tool can be applied.

[6]Different scene and action types categorized based on video content.

[7]Ablation studies are shown in Table V.

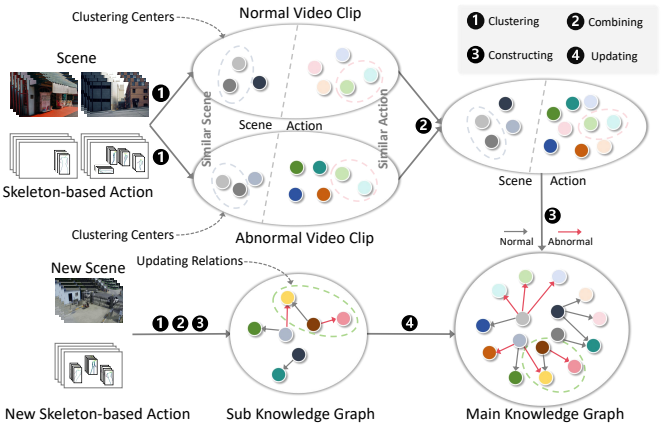

Fig. 5. Illustration of Relational Knowledge Mapper.

where $\mathbf{A}^{\text{fn}}$ and $\mathbf{A}^{\text{fa}}$ denote the cluster centers of the human actions from normal videos and abnormal videos, respectively, without considering if they are normal or abnormal actions. Here, $\cdot$ represents the dot product of the vectors, and $\| \ \|_2$ denotes the L2 norm of the vector.

Then, we combine the cluster centers of human actions from normal videos and abnormal videos — if the cosine similarity exceeds ρ[8], combining the two cluster centers. These cluster centers serve as the template to guide the subsequent knowledge graph construction. Note that the cluster centers of the scenes do not need to be combined.

3) Constructing: In a normal video, the occurrence of an action is always considered normal, whereas in an abnormal video, the occurrence of an action may not necessarily be abnormal; it could also be normal. Thus, as shown in Fig. 5-❸, to construct a detailed knowledge graph, we first use normal videos' scenes and human actions and mark these relationships as "normal". This serves as the initial knowledge graph.

Then, we incorporate abnormal videos' scenes and human actions into the initial knowledge graph. This is done by computing the cosine similarity between the human actions and the cluster centers in the initial knowledge graph, and based on this similarity, we assign a numerical identifier to the foreground. To achieve this process, we query the relationship between the scenes and human actions within the knowledge graph: if the relationship is "normal", we maintain it as is; if there is no relevant relationship, we mark it as "abnormal".

Let G represent the initial knowledge graph consisting of a number of scene-action relationships, denoted by $(\mathbf{S}, \mathbf{A}, \mathbf{R})$, where $\mathbf{S}$ and $\mathbf{A}$ are the cluster centers of the scenes and actions in normal videos, respectively, and $\mathbf{R}$ is the relation between scenes and actions of normal video clips:

$$G = \{(\mathbf{S}, \mathbf{A}, \mathbf{R})\}, \tag{5}$$

where $\mathbf{R}$ is defined as "normal" in the initial knowledge graph. In an unsupervised setting, G is the final knowledge graph G'; in a supervised setting, we can update the knowledge graph based on the relationships between scenes and human actions from abnormal videos.

$$G^{'} = \{(\mathbf{S}', \mathbf{A}', \mathbf{R}')\}, \tag{6}$$

where $\mathbf{S}'$ and $\mathbf{A}'$ denote the cluster centers of the scenes and actions contained within both normal and abnormal video clips. $\mathbf{R}'$ is the relationship between scenes and actions of normal and abnormal video clips. $\mathbf{R}'$ is defined as:

$$\mathbf{R}' = \begin{cases} Normal, & if \ \ (\mathbf{S}', \mathbf{A}', \mathbf{R}') \in G, \\ Abnormal, & if \ \ (\mathbf{S}', \mathbf{A}', \mathbf{R}') \notin G. \end{cases} \tag{7}$$

By querying and adjusting the relationships between scenes and human actions in the knowledge graph, these relationships can be effectively maintained or labeled as "normal" or "abnormal", resulting in the final knowledge graph $G^{'}$, providing support for Uncertainty Refinement (Sec. III-E).

4) Updating: If we want to add new video data that includes scenes and actions not previously included in $G^{'}$, we first need to construct a sub knowledge graph with the new data and then update the main knowledge graph, as illustrated in Fig. 5-❹. This updating process allows the knowledge graph to flexibly accommodate the inclusion of new data. This flexible knowledge graph updating mechanism provides the foundation for the system's continual learning and adaptation, enabling it to continuously adjust to evolving data and environments.

The updating process involves the dynamic generation of cluster centers based on the computation of cosine similarity between each newly added video data instance, *e.g.*, scenes and actions, and all scenes and actions cluster centers in the previously constructed knowledge graph, then, determine the maximum cosine similarity obtained, as outlined below:

$$\max\nolimits^{a}_{sim} = \mathrm{Max}\big(\bigcup\nolimits^{n}_{i} \mathrm{Sim}(\mathbf{A}^{\text{new}}_{i}, \mathbf{A}')\big), \tag{8}$$

$$\max\nolimits^{s}_{sim} = \mathrm{Max}\big(\bigcup\nolimits^{n}_{i} \mathrm{Sim}(\mathbf{S}^{\text{new}}_{i}, \mathbf{S}')\big), \tag{9}$$

where $\mathbf{A}^{\text{new}}_{i}$ and $\mathbf{S}^{\text{new}}_{i}$ are the newly added i-th action and scene. Sim denotes the cosine similarity. Max is the maximization operation to obtain the maximal value of cosine similarity of actions $(\max^{a}_{sim})$ and scenes $(\max^{s}_{sim})$. $\bigcup^{n}_{i}$ is the union of the values of cosine similarity. n means the total number of newly-added actions or scenes.

Based on the calculation results of the maximum cosine similarity, we add the newly added i-th action and scene as new cluster centers into $\mathbf{A}'$ and $\mathbf{S}'$, denoted as add.

$$\begin{cases} \mathbf{A}^{\text{new}}_{i} \overset{\text{add}}{\rightarrow} \mathbf{A}', & if \ \ \max\nolimits^{a}_{sim} \leq \mu_a, \\ \mathbf{S}^{\text{new}}_{i} \overset{\text{add}}{\rightarrow} \mathbf{S}', & if \ \ \max\nolimits^{s}_{sim} \leq \mu_s, \end{cases} \tag{10}$$

where μ_a and μ_s are thresholds to determine the add operation[9]. It's important to note that this process makes no distinction between normal and abnormal video clips.

Then, when the maximal value of cosine similarity of actions $(\max^{a}_{sim})$ and scenes $(\max^{s}_{sim})$ are greater than μ, we combine the newly-added i-th action and scene into $\mathbf{S}'$ and $\mathbf{A}'$, denoted by $combine$, with existing cluster centers in the constructed knowledge graph:

$$\begin{cases} \mathbf{A}^{\text{new}}_{i} \overset{\text{combine}}{\rightarrow} \mathbf{A}', & if \ \ \max\nolimits^{a}_{sim} > \mu_a, \\ \mathbf{S}^{\text{new}}_{i} \overset{\text{combine}}{\rightarrow} \mathbf{S}', & if \ \ \max\nolimits^{s}_{sim} > \mu_s. \end{cases} \tag{11}$$

Moreover, directly updating the main knowledge graph with all the relationships from the sub knowledge graph might lead

[8]The ablation study is shown in Table VII-A.

[9]The ablation study of these two thresholds can be seen in Table VIII.

to a decline or even failure in the model's detection capability, as there could be extreme or incorrect relationships in the sub knowledge graph. Therefore, we need to filter the relationships in the sub knowledge graph by calculating the cosine similarity between the nodes of the sub relationships and the nodes of the main relationships. If the sub relationship with the highest cosine similarity matches the main relationship, we proceed with the update; otherwise, we do not update the relationship. This ensures the safe updating of the main knowledge graph. It is important to note that all nodes in both the sub knowledge graph and the main knowledge graph come from $\mathbf{S}'$ and $\mathbf{A}'$.

During the inference phase, we only extract features (i.e., scene and action features) from the test set and match them with the pre-constructed RKM to infer anomaly scores. This entire feature extraction and matching process is entirely independent of any test set labels or ground truth information. The RKM functions as a static structure during inference, providing context-aware understanding of scene-action relationships learned from the training data.

In this way, we complete the construction of the detailed knowledge graph for "Relation Interweaving" to obtain scene-action relations. Next, we will detail how to use "Feature Interweaving" to obtain initial anomaly scores.

D. Scene-Action Integrator

As shown in Fig. 3-Step2, to improve the detection of human-related anomalies, we introduce a technique called the Scene-Action Integrator (SAI), which aims to effectively merge the features of actions and scenes. The SAI method not only focuses on individual motions and postures but also takes into account the environmental context. By gaining a deeper understanding of human movements and the meaning of the surrounding environment, it helps better analyze and interpret the relationship between human behavior and the environment, thus enhancing detection accuracy.

To implement the SAI, we use the decoupled scenes (sc) and the isolated human actions (sk) from the video clips. First, we use ResNet as the scene encoder to encode the scenes ($\mathcal{E}$). Similar to the method of processing action information in [11], we use a Graph Convolution Network (GCN) operation ($\mathcal{G}$) to capture semantic relationships. Positional embeddings ($\mathcal{PE}$) record the position (pos) of the actions in previous scenes, ensuring coherent integration and reasonable action arrangement when fusing with another action. To understand temporal dynamics, we employ a standard Long Short-Term Memory (LSTM) network ($\mathcal{LM}$) to process the actions and positions. By concatenating the features through the operation ($\mathcal{C}$) to obtain the fused features f_{concat}.This approach combines skeleton-based representations, semantic relationships, temporal dynamics, and positional information to generate accurate anomaly scores.

In the supervised setting, after being processed by a Fully Connected Layer (FC), the anomaly scores (AS) is finally generated. The entire processing process is expressed as follows:

$$\mathrm{AS} = \mathcal{FC}(\overbrace{f_{concat}}^{\Uparrow}). \\ \mathcal{C}(\mathcal{E}(sc), \mathcal{LM}(\mathcal{G}(sk)), \mathcal{LM}(\mathcal{PE}(pos))) \tag{12}$$

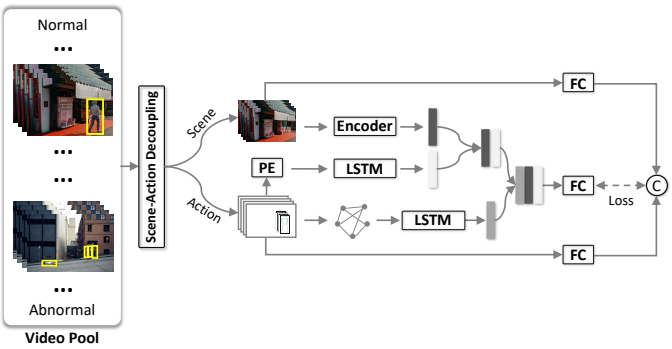

Fig. 6. Pipeline of unsupervised training, which is based on the traditional auto-encoder, using improved Scene-Action Integrator (Sec. III-D) as the backbone.

In the unsupervised setting, as shown in Fig. 6, we utilize SAI as the encoder (E_{SAI}) and construct a corresponding decoder (D_{SAI}) within this framework. The anomaly score (AS) is obtained by comparing the combined features of the input video with the reconstruction error as follows:

$$\mathrm{AS} = MSE(f_{concat}, D_{SAI}(E_{SAI}(f_{concat}))), \tag{13}$$

where MSE refers to Mean Squared Error.

SAI as the backbone model of DecoAD, with the ability to directly perform video anomaly detection. To further enhance DecoAD's detection capability and fully exploit the advantages of "relation interweaving" and "feature interweaving", we designed the Uncertainty Refinement (UR) module.

E. Uncertainty Refinement

To enhance the model's ability to handle ambiguous or borderline cases and improve detection accuracy, we propose the Uncertainty Refinement(UR), which iteratively trains our DecoAD in Step 3[10]. In this module, we define two hyperparameters, β_1 and β_2, as thresholds[11] to guide the classification of video segments into three pools, $i.e.$, "normal pool", "abnormal pool" and "pending pool". For the video segments in the training set, all scene information, including positional data, is combined with human actions and fed into the model trained in Step 2. During the first iteration, video segments are distributed into these three pools based on their anomaly scores and relationships derived from the knowledge graph ($G^{'}$): 1) Video segments with anomaly scores below β_1 and labeled as "normal" in the knowledge graph $G^{'}$ are placed into the Normal Pool as normal training samples. 2) Video segments with anomaly scores above β_2 and not labeled as "normal" in the knowledge graph $G^{'}$ are placed into the Abnormal Pool as abnormal training samples. 3) Video segments that do not meet the above two conditions are placed into the Pending Pool as borderline or uncertain samples requiring further refinement.

Video segments in the Pending Pool represent ambiguous samples that the model struggles to classify with confidence. These samples are not directly used for training but are iteratively fed back into the model for continuous optimization. During each iteration, the model trains using the definitive

[10]The ablation study is shown in Table VII-C.
[11]The ablation study is shown in Table VI.

samples from the Normal Pool and Abnormal Pool, dynamically adjusting its decision boundaries, particularly for samples near classification thresholds. This process enhances the model's ability to handle borderline cases and optimizes anomaly scores. In this process, the Normal Pool provides definitive normal samples, while the Abnormal Pool supplies definitive abnormal samples. Together, these two pools serve as the foundation for iterative model optimization, enabling the model to more accurately distinguish between normal and abnormal behaviors. Through repeated iterations, ambiguous samples from the Pending Pool are continually fed back into the model for calibration, resulting in dynamic adjustment and sustained optimization of decision boundaries.

By focusing on these ambiguous samples, the UR effectively reduces false positives and false negatives, significantly improving the model's performance on borderline samples. Additionally, this iterative optimization method enhances the model's adaptability to new scenarios, ensuring robust performance across diverse environments. Ultimately, the UR successfully addresses ambiguity in the anomaly detection process, reduces misclassification rates, and comprehensively improves the overall accuracy and robustness of the DecoAD.

F. Training Loss

The training of DecoAD is divided into two stages. In the first stage, the original data is used to directly train SAI. In the second stage, all scenes and actions in RKM are recombined, and SAI is utilized to score all possible combinations. New training samples are then generated based on the rules in UR, and SAI undergoes a second round of training.

In the first stage. During the training phase of SAI, for both fully-supervised and weakly-supervised settings, we calculate the Multiple Instance Learning loss ($\mathcal{L}_{s_1}$) [50] by comparing the anomaly scores of abnormal and normal videos. The design of Multiple Instance Learning (MIL) enables the model to focus on the most discriminative key instances (clips[12]) within videos, thereby better capturing contextual information and the complex interdependencies between scenes and actions. This characteristic allows the model to adapt more effectively to various scenes and behavior patterns. Although fully-supervised settings provide frame-level labels, real-world datasets may still contain noisy annotations or ambiguous boundary samples. By focusing on the most discriminative instances within positive and negative bags, MIL effectively mitigates the adverse effects of noisy labels during training, improving the model's robustness and classification performance. The entire process can be formulated as follows:

$$\mathcal{L}_{s_1} = \alpha_1 \times \mathcal{L}_{rank} + \alpha_2 \times \mathcal{L}_{focal}, \tag{14}$$

where α_1 and α_2 are learnable weight parameters. $\mathcal{L}_{focal}$ is the Focal Loss [51] incorporating with BCE Loss. $\mathcal{L}_{rank}$ is the Ranking Loss [52], which can be denoted by:

$$\mathcal{L}_{rank} = \max(0, 1 + S_{\mathrm{n}} - S_{\mathrm{a}}), \tag{15}$$

where $\mathcal{S}_n$ and $\mathcal{S}_a$ respectively represent the means of the topK[13] scores of normal video segments and abnormal video segments.

For unsupervised training, the loss ($\mathcal{L}_{s_1}$) for unsupervised training are consisting of reconstruction loss ($\mathcal{L}_{rec}$) and regularization term ($\mathcal{L}_{reg}$) is formulated as:

$$\mathcal{L}_{s_1} = \lambda_1 \times \mathcal{L}_{rec} + \lambda_2 \times \mathcal{L}_{reg}, \tag{16}$$

where λ_1 and λ_2 are learnable weight parameters. The regularization term $\mathcal{L}_{reg}$ is calculated using L2 regularization to prevent overfitting by penalizing large weights in the model.

In the second stage. We employ the Binary Cross-Entropy loss to increase the distance between the "normal pool" and the "abnormal pool". The total loss ($\mathcal{L}_{s_2}$) is formulated as:

$$\mathcal{L}_{s_2} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)], \tag{17}$$

where N represents the number of samples. y_i represents the true label that the i-th sample and $\hat{y}_i$ represents the predicted probability that the i-th sample.

IV. Experiments

A. Datasets

We evaluate our method on four datasets, namely NWPU Campus [53], UBnormal [27], ShanghaiTech Campus [54] and UFSR. According to the characteristics of each dataset, we use the UBnormal dataset for fully-supervised training, the NWPU Campus, UFSR and UBnormal datasets for weakly-supervised training, and the NWPU Campus, UFSR, UBnormal and ShanghaiTech Campus dataset for unsupervised training.

The NWPU Campus dataset contains 43 different scenes and 28 types of abnormal events, pioneering the research on scene-related anomalies. However, its training set only includes normal video data, which does not meet the requirements of weakly-supervised video anomaly detection. Therefore, we adopted the weakly-supervised NWPU Campus dataset reconfigured in [55], but still used the original dataset for unsupervised training. The UBnormal dataset consists of 29 scenes and 22 types of abnormal events, with detailed annotations, making it extremely valuable for advanced anomaly detection research. The ShanghaiTech Campus dataset focuses on campus scenes, covering 13 scenes and 11 types of abnormal events.

Most of the existing video anomaly detection datasets mainly focus on static scenes. To further verify the ability of our method to handle dynamic scenes, we constructed a new dataset named UFSR. The UFSR dataset is based on the UBI-Fights dataset [56], which contains videos of fighting scenes in dynamic environments but lacks scene-related anomalies. We incorporated the fighting videos in legitimate settings as normal video samples (e.g., boxing). Therefore, in the UFSR dataset, fighting is defined as a scene-related anomaly. As far as we know, the UFSR dataset is the first one that is designed for dynamic scenes and features scene-related anomalies.

[12]We compile N clips from each normal video into a normal bag, while N clips from an abnormal video are grouped into an abnormal bag. Each clip contains 24 frames. The ablation study is shown in Table VII-B.

[13]In the weakly-supervised setting, we select the topK = 4 samples within each bag for training, while in the fully-supervised setting, topK = batch size / 2. This ensures the adaptability of MIL across different supervision settings.

B. Evaluation Metrics

In the field of video anomaly detection, the commonly used performance evaluation metric is the area under the Receiver Operating Characteristic curve (AUC), which intuitively reflects the performance of detection methods. However, due to the imbalance in anomaly detection tasks, AUC may exaggerate performance. Therefore, we introduce the area under the Precision-Recall curve (AP) as a supplementary metric. A higher AP value indicates the model's stronger ability to detect abnormal events.

C. Implementation Details

Our work is implemented in PyTorch and experimented on NVIDIA RTX 4090 GPU. We employ the AlphaPose [46] and YOLOX [57] detectors to independently detect the human skeleton in each video frame. The network is optimized using the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with an initial learning rate of 1×10^{-2} for all model training, which decreases by multiplying 0.8 for every 10 epochs. Our method utilizes a batch size of 256, and the training process runs for a total of 120 epochs, only costing 2.2 hours. The size of our supervised model has been optimized to 1.0 Mb, while the unsupervised model size has been optimized to 12.3 Mb.

D. Component Evaluation

We conducted a comprehensive evaluation of our method's components, as shown in Table I. To ensure successful code execution, we replaced the key components requiring verification with simpler operations. For example, we substituted the proposed components with a basic ResNet model [58] consisting of two fully connected layers. This served as our baseline, and the qualitative results are shown in line 1.

Lines 2-5 demonstrate the effectiveness of the Scene-Action Integrator (Sec. III-D) in achieving "Feature Interweaving" between scenes and associated human actions. Comparing line 4 to line 11, where we removed LSTM and GCN, we observed a decrease in AUC from 78.4% to 72.6%. Additionally, we observed that line 3 (GCN) outperformed line 2 (LSTM), with AUC values of 65.9% and 71.6%, respectively, indicating that GCN is better at modeling action relationships, which is crucial for understanding human actions. These results underscore the importance of the Scene-Action Integrator in capturing the relationship between scenes and human actions, and highlight the effectiveness of GCN in this task.

Lines 6-9 provide evidence of the effectiveness of Uncertainty Refinement (Sec. III-E). By comparing line 7 to line 8, we deduced that the iterative training process of the "pending" pool is more effective than using binary cross-entropy (BCE) loss for the "normal" pool and "abnormal" pool, as indicated by the higher AUC. Moreover, removing the two constraints on anomaly score and scene-action relation (line 9) resulted in decreased AUC performance.

Comparing line 10 to line 11, our method incorporating the Relational Knowledge Mapper (Sec. III-C, line 11) outperforms the method without it (line 10). This is because the Relational Knowledge Mapper enables a comprehensive understanding of the intricate interplay between different scenes and human actions by leveraging a detailed knowledge graph.

TABLE I
Quantitative evaluation of major components used in our approach in terms of the AUC and AP performance on the UBnormal (UB) dataset. The best results are marked in **bold**.

	Major Components: SAI			UR			RKM	Dataset: UB	
	LSTM	GCN	PE	Iter	BCE	2CoS	KG	AUC	AP
0 1	✘	✘	✘	✘	✘	✘	✘	0.634	0.690
2	✘	✘	✔	✔	✔	✔	✔	0.659	0.698
1 3	✘	✔	✔	✔	✔	✔	✔	0.716	0.745
4	✔	✘	✔	✔	✔	✔	✔	0.726	0.776
5	✔	✔	✘	✔	✔	✔	✔	0.778	0.815
6	✔	✔	✔	✘	✘	✔	✔	0.768	0.809
2 7	✔	✔	✔	✔	✘	✔	✔	0.774	0.815
8	✔	✔	✔	✘	✔	✔	✔	0.771	0.812
9	✔	✔	✔	✔	✔	✘	✔	0.773	0.810
3 10	✔	✔	✔	✔	✔	✔	✘	0.772	0.818
4 11	✔	✔	✔	✔	✔	✔	✔	**0.784**	**0.824**

0 Baseline 1 *Verify* SAI 2 *Verify* UR 3 4 *Verify* RKM

SAI: Scene-Action Integrator (Sec. III-D) **PE:** Position Embedding
UR: Uncertainty Refinement (Sec. III-E) **Iter:** Iteration
RKM: Relational Knowledge Mapper (Sec. III-C) **KG:** Knowledge Graph
2CoS: Two Constrains -- anomaly score and scene-action relation

E. Performance Comparison

To demonstrate the effectiveness of our approach, we conducted a comprehensive comparison with state-of-the-art methods using three different training methodologies: fully-supervised, weakly-supervised, and unsupervised training.

For fully/weakly-supervised training, we selected the Deep-MIL [2], RTFM [4], MGFN [3], RTFM-BERT [5], BN-WVAD [30], TDSD [55], Joint-RTFM [59], ST-GCN [9], Shift-GCN [10], STG-NF [11]. For unsupervised training, we evaluated the MPN [60], LGN-Net [61], CampusVAD [53], LAVAD [38], SSAE [62], GEPC [63], MoCoDAD [31], STG-NF [11], TrajREC [32], Joint-STG-NF [59] and HSC [64] methods. The results we compared were obtained either from the source code or reported results provided by the respective authors. "Ours[1]" refers to one of our methods that uses only skeleton information for video anomaly detection. "Ours[2]" is our method, which only uses the Scene-Action Integrator (SAI) component for anomaly detection to prove the effectiveness of "Scene-Action Interweaving". "Ours[3]" is also our method, which uses scene data for training without removing action information. For details, please refer to Sec. III-B2. "Ours*" comprehensively considers all information (skeleton, scene, and location).

1) Quantitative Comparisons with Fully/Weakly-supervised Training Methods: The quantitative comparison results with fully/weakly-supervised training methods are shown in Table II. We found that "Ours[1]" shows inferior performance compared to existing action-based methods such as STG-NF. STG-NF overlooks scene information, operating directly on the distribution of data and providing a more direct probabilistic interpretation, making it more sensitive to the detection of abnormal behaviors. Our proposed method "Ours*" outperforms all previous state-of-the-art approaches in fully/weakly-supervised training settings. Specifically, "Ours*" achieves an

TABLE II
Quantitative performance comparison with other methods on the NWPU Campus (NWPUC), UFSR, and UBnormal (UB) datasets in terms of AUC and AP metrics under fully and weakly-supervised training (denoted as "fully" and "weakly," respectively). Red represents the best performance, green represents the second best. "Type" represents the feature dependency: "A" for appearance and "S" for action (skeleton).

Model	Model Size	Type	NWPUC (weakly)		UFSR (weakly)		UB (weakly)		UB (fully)	
			AUC	AP	AUC	AP	AUC	AP	AUC	AP
DeepMIL$_{18}$	8.5MB	A	0.656	0.071	0.645	0.269	0.552	0.622	-	-
RTFM$_{21}$	50.7MB	A	0.707	0.109	0.711	0.307	0.645	0.676	-	-
MGFN$_{22}$	114.7MB	A	0.709	0.082	0.720	0.304	0.557	0.590	-	-
RTFM-BERT$_{24}$	129.3MB	A	0.719	0.094	0.721	0.340	0.643	0.671	-	-
BN-WVAD$_{24}$	23.2MB	A	0.737	0.093	0.752	0.421	0.685	0.730	-	-
TDSD$_{24}$	-	A	0.802	-	-	-	-	-	-	-
Joint-RTFM$_{25}$	98.9MB	A	0.720	0.096	0.724	0.347	0.642	0.683	-	-
ST-GCN$_{18}$	0.4MB	S	0.683	0.066	0.556	0.210	0.729	0.771	0.745	0.787
Shift-GCN$_{20}$	0.6MB	S	0.660	0.057	0.546	0.195	0.667	0.726	0.678	0.734
STG-NF$_{23}$	0.2MB	S	0.658	0.063	0.544	0.201	0.753	0.786	0.792	0.824
Ours$^{1}_{(Skeleton)}$	0.5MB	S	0.635	0.097	0.566	0.185	0.701	0.743	0.711	0.745
Ours$^{2}_{(SAI)}$	1.0MB	A+S	0.721	0.101	0.716	0.434	0.763	0.795	0.781	0.812
Ours$^{3}_{(Noise)}$	1.0MB	A+S	0.749	0.132	0.745	0.476	0.777	0.822	0.785	0.823
Ours*$_{(All)}$	1.0MB	A+S	0.764	0.155	0.763	0.511	0.784	0.824	0.803	0.834

TABLE III
Quantitative performance comparison with other methods on the NWPU Campus (NWPUC), UFSR, UBnormal (UB), and ShanghaiTech Campus (STC) datasets, regarding AUC and AP metrics in unsupervised training (denoted as "un"). Red color represents the best, and green color represents the second best. "Type" represents the feature dependency: "A" for appearance and "S" for action (skeleton).

Model	Model Size	Type	NWPUC (un)		UFSR (un)		UB (un)		STC (un)	
			AUC	AP	AUC	AP	AUC	AP	AUC	AP
MPN$_{21}$	159.5MB	A	0.562	0.195	0.489	0.180	0.546	0.566	0.692	0.610
LGN-Net$_{22}$	91.1MB	A	0.572	0.214	0.451	0.158	0.559	0.585	0.679	0.594
CampusVAD$_{23}$	-	A	0.682	-	-	-	-	-	-	-
LAVAD$_{24}$	13.5GB	A	0.514	0.178	0.573	0.255	0.590	0.656	0.479	0.412
SSAE$_{24}$	-	A	0.756	-	-	-	-	-	-	-
GEPC$_{20}$	3.6MB	S	0.681	0.220	0.578	0.258	0.516	0.557	0.721	0.601
MoCoDAD$_{23}$	2.0MB	S	0.657	0.250	0.435	0.148	0.688	0.695	0.745	0.655
STG-NF$_{23}$	0.2MB	S	0.661	0.160	0.524	0.183	0.718	0.769	0.859	0.815
TrajREC$_{24}$	0.02MB	S	0.675	0.268	0.570	0.204	0.662	0.684	0.743	0.697
Joint-STG-NF$_{25}$	0.2MB	S	0.636	0.226	0.531	0.189	0.709	0.767	0.806	0.764
HSC$_{23}$	-	A+S	-	-	-	-	-	-	0.834	-
Ours$^{1}_{(Skeleton)}$	11.8MB	S	0.663	0.260	0.568	0.171	0.676	0.746	0.722	0.652
Ours$^{2}_{(SAI)}$	12.3MB	A+S	0.684	0.315	0.631	0.269	0.735	0.774	0.816	0.770
Ours$^{3}_{(Noise)}$	12.3MB	A+S	0.674	0.323	0.647	0.286	0.715	0.759	0.821	0.753
Ours*$_{(All)}$	12.3MB	A+S	0.687	0.335	0.660	0.301	0.739	0.784	0.837	0.775

improvement of 1.1% and 3.1% in AUC values, and 9.0% and 3.8% in AP values over the best existing weakly-supervised methods on UFSR and UBnormal, respectively. Moreover, it achieves an improvement of 1.1% in AUC value and 1.0% in AP value over the best existing fully-supervised method on UBnormal. These results demonstrate the effectiveness of our proposed method, which leverages the "Scene-Action Interweaving" approach to combine and analyze elements from different scenes and human actions in videos for enhanced anomaly detection.

2) Quantitative Comparisons with Unsupervised Training Methods: The quantitative comparison results with unsupervised training methods are shown in Table III. We found that "Ours^{3}" performs worse than "Ours*" because the use of scene data containing action information interfered with the model's training, thereby affecting its performance. Our "Ours*" method also surpasses all previous state-of-the-art unsupervised training methods in UFSR and UBnormal. "Ours*" achieves improvements of 8.2% and 2.1% in AUC values,

TABLE IV
Comparison of generalization performance across methods, where models are trained on the NWPU Campus (NWPUC), UBnormal (UB), and NWPUC+UB datasets under weakly-supervised settings, and evaluated on the NWPUC dataset in terms of AUC and AP metrics. The best results are highlighted in **bold**.

	Model	Training on NWPUC		Training on UB		Training on NWPUC+UB	
		Testing on NWPUC (AUC/AP)					
Weakly-supervised	DeepMIL$_{18}$	0.656	0.071	0.531	0.048	0.673	0.081
	RTFM$_{21}$	0.707	0.109	0.515	0.068	0.701	0.097
	MGFN$_{22}$	0.709	0.082	0.525	0.047	0.713	0.089
	RTFM-BERT$_{24}$	0.719	0.094	0.540	0.044	0.731	0.101
	BN-WVAD$_{24}$	0.737	0.093	0.559	0.053	0.742	0.105
	Joint-RTFM$_{25}$	0.720	0.096	0.575	0.048	0.723	0.091
	ST-GCN$_{18}$	0.683	0.066	0.628	0.053	0.684	0.073
	Shift-GCN$_{20}$	0.660	0.057	0.642	0.052	0.654	0.057
	STG-NF$_{23}$	0.658	0.063	0.643	0.055	0.695	0.078
	Ours*	**0.764**	**0.155**	**0.692**	**0.086**	**0.770**	**0.163**
Unsupervised	MPN$_{21}$	0.562	0.195	0.523	0.177	0.541	0.203
	LGN-Net$_{22}$	0.572	0.214	0.565	0.212	0.559	0.207
	GEPC$_{20}$	0.681	0.220	0.665	0.213	0.677	0.206
	MoCoDAD$_{23}$	0.657	0.250	0.630	0.224	0.665	0.267
	STG-NF$_{23}$	0.661	0.160	0.648	0.154	0.623	0.183
	TrajREC$_{24}$	0.675	0.268	0.646	0.237	0.655	0.198
	Joint-STG-NF$_{25}$	0.636	0.226	0.622	0.223	0.638	0.221
	Ours*	**0.687**	**0.335**	**0.654**	**0.263**	**0.696**	**0.337**

and 4.3% and 1.5% in AP values over the best existing unsupervised method, MoCoDAD, on the UFSR and UBnormal datasets, respectively. Although some methods have smaller model sizes, their video anomaly detection capabilities are not excellent. Our method (both supervised and unsupervised), after balancing model size and video anomaly detection capability, achieves the best performance. Finally, "Ours*" significantly outperforms "Ours^{2}", demonstrating that RKM and UR are effective not only in weakly-supervised and fully-supervised settings, but also in unsupervised settings.

3) Qualitative Results: Fig. 7 demonstrates the superior results of our method in context-related (i.e., scene-related) scenarios. We visualize anomaly scores and compare them with the appearance-based method BN-WVAD [30] and the action-based method STG-NF [11]. Our method successfully and promptly detects abnormal events by generating high anomaly scores for abnormal frames. Notably, in the case of D235_07, a person is riding a bicycle in a square where cycling is prohibited, while in D235_20, a person is cycling on the road. The former is an abnormal event, and the latter is a normal one. Our model successfully identifies and detects the abnormal event in the scene without any false alarms, thanks to the concept of "Scene-Action Interweaving".

4) Generalization Performance: We trained the model using weakly-supervised and unsupervised learning on the NWPU Campus dataset, UBnormal dataset, and a mixed dataset of both, and tested it on the NWPU Campus dataset to validate its ability to handle "new" or "unknown" scenes and dynamic scaling. We compared our method with appearance-based and action-based approaches. The experimental results show that our proposed "Ours*" method demonstrates strong generalization capability. Under weakly-supervision, when trained and tested on the NWPU Campus dataset, the AUC and AP are 76.4% and 15.5%, respectively; when trained on the UBnormal dataset and tested on NWPU Campus, the AUC and AP are 69.2% and 8.6%; and when trained on the

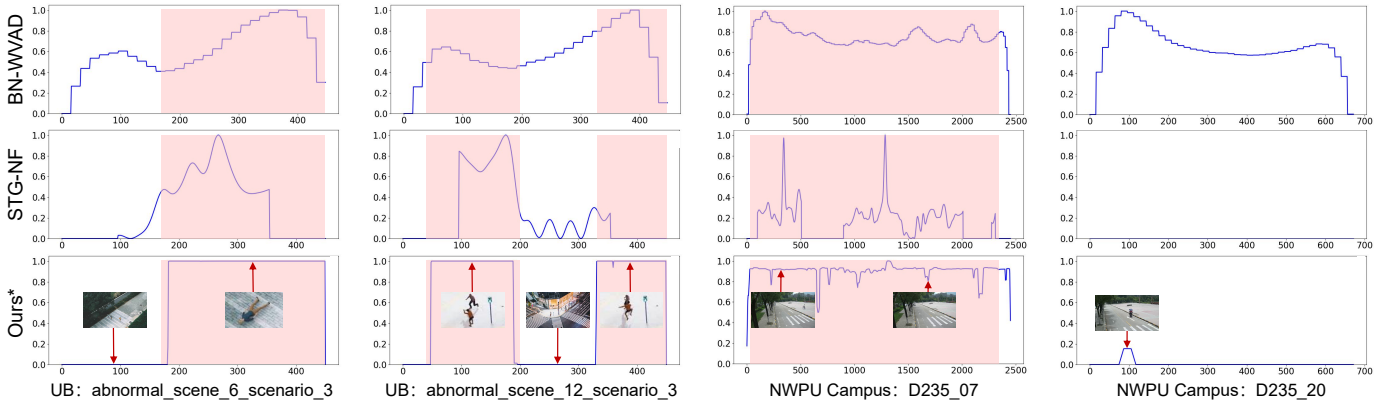

Fig. 7. The qualitative comparison of our weakly-supervised method with BN-WVAD and STG-NF on the testing videos. Colored windows indicate the true abnormal regions. Enlarging the view enhances the effectiveness.

TABLE V
Ablation study on different clustering center numbers (Sec. III-C1) on the UBnormal dataset. "θ_{fn}" and "θ_{fa}": clustering center number of human actions within normal and abnormal video clips.

θ_{fa} \ θ_{fn}	5	10	15	20	25
15	0.779	0.781	0.778	-	-
20	0.776	0.777	0.777	0.780	-
25	0.782	0.781	**0.784**	0.783	0.782
30	0.779	0.780	0.781	0.780	0.782
35	0.781	0.781	0.782	0.779	0.780

TABLE VI
Ablation study on different thresholds for constructing three pools (Sec. III-E) on the UBnormal dataset. "β_1" and "β_2": different thresholds used to divide the three pools.

β_2 \ β_1	0.1	0.2	0.3	0.4	0.5
0.5	0.779	0.780	0.780	0.781	0.781
0.6	0.778	0.779	0.781	0.782	0.780
0.7	0.778	0.780	0.781	0.783	0.781
0.8	0.780	0.781	0.781	**0.784**	0.782
0.9	0.779	0.779	0.780	0.782	0.779

mixed dataset, the AUC and AP are 77.0% and 16.3%. Under unsupervised learning, when trained and tested on the NWPU Campus dataset, the AUC and AP are 68.7% and 33.5%; when trained on the UBnormal dataset and tested on NWPU Campus, the AUC and AP are 65.4% and 26.3%; and when trained on the mixed dataset, the AUC and AP are 69.6% and 33.7%. It is noteworthy that appearance-based methods, due to their reliance on low-level pixel features, exhibit poor generalization and a significant drop in performance when the scene changes. In contrast, our method effectively models the complex relationship between scenes and actions, reducing reliance on specific visual features, enhancing the model's adaptability, and ensuring robust generalization across different datasets.

F. Ablation Study

1) Choices of the Number of Cluster Centers: The clustering operation in the "Relational Knowledge Mapper" (see Sec. III-C1) aims to unify similar scenes and actions into the same category, thereby simplifying scene complexity and

TABLE VII
Ablation study on different cosine similarity thresholds for fusing two clustering centers (**A**) (Sec. III-C2), different segment lengths (**B**) (Sec. III-D), and different iteration times (**C**) (Sec. III-E). "ρ": cosine similarity threshold; "f": video clip frame numbers; "t": iteration times of the Step3; NWPUC represents the NWPU Campus dataset, UB represents the UBnormal dataset, STC represents the ShanghaiTech Campus dataset.

A

Sets	UB
ρ = 0.70	25
ρ = 0.80	25
ρ = 0.85	25
ρ = 0.90	26
ρ = 0.95	**30**

B

Sets	NWPUC	UB	STC
f = 12	0.645	0.721	0.795
f = 16	0.666	0.730	0.811
f = 20	0.660	0.733	0.819
f = 24	**0.687**	**0.739**	**0.837**
f = 30	0.679	0.719	0.825

C

Sets	UB
t = 4	0.774
t = 6	0.778
t = 8	0.781
t = 10	**0.784**
t = 12	0.782

TABLE VIII
Ablation study on the updating cosine similarity thresholds μ_a for actions and μ_s for scenes (Sec. III-C); UB represents the UBnormal dataset.

A

Sets	UB
μ_a = 0.30	13
μ_a = 0.35	15
μ_a = 0.40	20
μ_a = 0.45	**27**
μ_a = 0.50	34

B

Sets	UB
μ_s = 0.75	15
μ_s = 0.80	18
μ_s = 0.85	26
μ_s = 0.90	**29**
μ_s = 0.95	29

reducing the number of categories. We conducted an ablation study on the UBnormal dataset. As shown in Table V, when the number of cluster centers is too small, the model struggles to effectively distinguish between similar scenes and actions, reducing its detection performance. Conversely, if the number of cluster centers is too large, the learned features become overly granular, which affects generalization. We recommend setting the number of cluster centers for human actions in normal and abnormal video segments to 15 and 25, respectively, to achieve a balance between coverage and discrimination.

2) Choices of β_1 and β_2 in Constructing Three Pools: We further conducted experiments on the UBnormal dataset, as shown in Table VI, to explore the impact of different thresholds on the classification of the "pending pool" (see Sec. III-E). Setting the thresholds too high or too low can result in inappropriate sensitivity of the model to the data. When β_1 is set too low, normal video clips may be incorrectly classified as abnormal; if β_2 is set too high, abnormal data may be

TABLE IX
Ablation studies evaluate the impact of segmentation, inpainting tools, and skeletal extraction accuracy on model performance, assessed on the UBnormal dataset under weakly-supervised settings using AUC and AP metrics, with the best results highlighted in **bold**.

Sets	None		MRCNN+IAM		SAM+IAM		AILab	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP
ResNet_{50}	0.735	0.776	0.734	0.782	0.741	0.787	0.737	0.788
HRNet_{32}	0.737	0.774	0.740	0.785	0.744	0.784	0.739	0.783
ResNet_{152}	0.777	0.812	0.779	0.819	**0.784**	**0.824**	0.782	0.823

mistakenly classified as normal, thereby reducing the overall performance of DecoAD. We recommend setting β_1 and β_2 to 0.4 and 0.8, respectively, to ensure effective differentiation between normal and abnormal data in the "pending pool".

3) Choices of Clustering Threshold ρ, Video Segment Length, and Number of Iterations: We conducted ablation studies to evaluate the impact of key parameter settings on model performance, as shown in Table VII. For the cosine similarity threshold ρ in combining cluster centers (see Sec. III-C2), setting ρ to 0.95 yielded results closest to the actual number of categories. Next, we assessed the effect of video segment length (see Sec. III-D) and found that a segment length of 24 frames provided the best balance between data completeness and processing complexity. Finally, we tested the number of iterations in the uncertainty refinement process (see Sec. III-E). While performance improved with more iterations, it stabilized after 10, likely due to insufficient data in the "pending pool", making it difficult to further expand the "normal pool" and "abnormal pool", and the model may have already converged.

4) Effectiveness of the Updating Thresholds: We further conducted an ablation study on the updating thresholds μ_a (for actions) and μ_s (for scenes) (see Sec. III-C). To determine the updating thresholds, we carried out ablation experiments on the same dataset. As shown in Table VIII, we found that when μ_a was set to 0.45, the number of action clusters was closest to the actual number of action categories. Similarly, when μ_s was set to 0.90, the number of scene clusters was closest to the actual number of scene categories, indicating that the updating effect was optimal at these thresholds.

5) Different Scene-Action Decoupling Tools: To verify whether the choice of scene-action decoupling tools affects the accuracy of the model, we conducted an ablation study on different image segmentation tools, image inpainting tools, and different structures of AlphaPose (i.e., ResNet50, HRNet32, ResNet152), as shown in Table IX. In these experiments, MRCNN (Mask R-CNN) and SAM are different image segmentation tools, while IAM and AILab (AILab is an online tool that requires manually generating masks) are different image inpainting tools. "None" refers to no image segmentation or inpainting operations being performed. The experimental results show that the selection of image segmentation and inpainting tools does affect the results, but the impact of not performing any processing is more significant, as it leads to interference from behavioral noise. Additionally, the use of different AlphaPose structures (i.e., varying quality of skeleton extraction) has a significant impact on the model's performance.

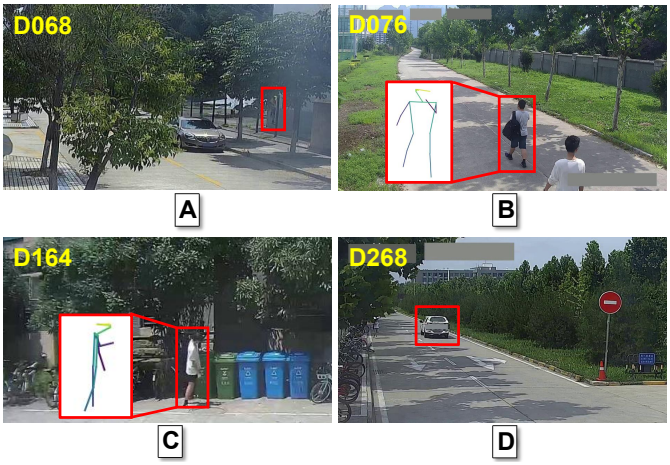

Fig. 8. Failure cases on the NWPU Campus dataset. The yellow labels in the top-left corner represent the scenario IDs.

G. Privacy Implications and Ethical Considerations

In the deployment of video anomaly detection systems, especially in surveillance scenarios, privacy protection and ethical considerations are of paramount importance. These systems inevitably involve the processing of sensitive data, which may include personally identifiable information or individual behavioral patterns. Traditional video anomaly detection systems typically rely on raw video footage, which inherently contains identifiable visual information, such as facial features, clothing details, and other personal identifiers. Handling such data in public surveillance scenarios can pose significant privacy risks, potentially leading to data misuse. To mitigate these risks, our approach utilizes skeleton data extracted from raw video footage.

Skeleton data abstracts human motion into key points and connections, effectively removing identifiable facial and body details. This abstraction preserves sufficient behavioral feature information to support anomaly detection while significantly reducing the risk of exposing sensitive personal information. Furthermore, the non-visual nature of skeleton data ensures that even in the event of a data breach, the potential for misuse is greatly minimized.

H. Limitations

Although DecoAD shows promise in addressing the limitations of current human video anomaly detection methods, our analysis of the NWPU Campus dataset (see Fig. 8) revealed several limitations: 1) It is difficult to distinguish between very similar behaviors, especially when combined with the surrounding context; 2) In complex scenarios with occlusions or background distractions, the skeleton extraction may be incomplete, affecting the accuracy of anomaly detection; 3) Skeleton-based data cannot detect anomalies related to visual aspects, such as improper clothing; 4) The method struggles to detect non-human-related anomalies, such as traffic violations by vehicles. Additionally, our approach relies on auxiliary tools like skeleton estimation, object segmentation, and image inpainting. While these tools improve performance, they also introduce additional complexity and dependencies. We conducted a detailed time analysis for each step (see

TABLE X
Detailed average time cost for processing a single video frame. This result was obtained on a PC equipped with an Intel(R) Xeon(R) CPU and an NVIDIA GTX 4090 GPU (with 24G RAM). The experiment was conducted on an SSD set.

Main Steps	FPS	Milliseconds
Key Comp. 1: Scene-Action Decoupling (Sec. III-B2)		**81.38350ms**
1) *Skeleton Extraction*		39.29161ms
2) *Image Segmentation*		22.95294ms
3) *Image Inpainting*		19.13895ms
(The processing time of 2) and 3) in static scenes can be neglected)		
Key Comp. 2: Scene-Action Integrator (Sec. III-D)		**2.82666ms**
1) *Action Feature Processing*		2.81971ms
2) *Scene Feature Processing*		0.00425ms
3) *Position Feature Processing*		0.00007ms
4) *Feature Fusion*		0.00263ms
Key Comp. 3: Relational Knowledge Mapper (Sec. III-C)		**22.98114ms**
(Key Comp. 3 is only used for the training phase)		
Total Inference Time (static scene)	**23.7**	**42.11827ms**
Total Inference Time (dynamic scene)	**11.9**	**84.21016ms**

Table X). Notably, image segmentation and inpainting require more processing time. However, for static scenes, only a single scene image needs to be processed offline, making the time overhead negligible. For dynamic scenes, we optimize efficiency by processing one scene image per video segment (e.g., every 24 frames). As a result, the frames per second (FPS) reaches 23.7 in static scenes and 11.9 in dynamic scenes, ensuring practical usability.

Furthermore, we evaluated the model's performance without using image segmentation and inpainting (refer to "Ours^2" in Tables II and III). The results indicate that our method still achieves competitive performance, highlighting its robustness even when auxiliary tools are excluded. Future improvements to these auxiliary tools are expected to further enhance processing speed.

V. Conclusion

This study proposes DecoAD, an innovative architecture dedicated to detecting anomalies in human-related videos. The core innovation of DecoAD lies in the decoupling and interweaving of scenes and actions, successfully integrating appearance-based and action-based features. In the decoupling stage, the scene and action features are precisely separated from the video data, enabling the model to independently model the characteristics of both without mutual interference. Subsequently, in the interweaving process, through carefully constructed knowledge graphs and other means, the complex relationships between the scene and action features are deeply explored and reconstructed, thus achieving the semantic integration of the two. This unique processing method enables DecoAD to perform excellently in human-related video anomaly detection. A large number of experimental results show that our model has significant detection advantages and extremely strong generalization ability.

An important direction for future research is to deeply explore the potential of the model to integrate appearance features while maintaining privacy protection. Under the premise of ensuring privacy security, the reasonable integration of appearance features and skeleton features is expected to bring multiple improvements to the system.

Acknowledgments: This work was supported in part by the National Natural Science Foundation of China under Grant 62172246, in part by Excellent Young Scientists Fund of Natural Science Foundation of Shandong Province under Grant ZR2024YQ071, in part by the Key Laboratory of Forensic Examination for Sichuan Provincial Universities under Grant 2024YB01, in part by the Postgraduate Innovation Fund of China University of Petroleum (East China) under Grant 24CX04028A, and in part by the Fundamental Research Funds for the Central Universities under Grant 22CX06037A.

References

[1] S. Yuan, L. Li, N. Yu, T. Peng, X. Hu, and X. Pan, "Anomaly detection of industrial products considering both texture and shape information," in *CGIC*. Springer, 2023, pp. 149–160.
[2] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *CVPR*, 2018, pp. 6479–6488.
[3] Y. Chen, Z. Liu, B. Zhang, W. Fok, X. Qi, and Y.-C. Wu, "Mgfn: Magnitude-contrastive glance-and-focus network for weakly-supervised video anomaly detection," in *AAAI*, vol. 37, no. 1, 2023, pp. 387–395.
[4] 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 *ICCV*, 2021, pp. 4975–4986.
[5] W. Tan, Q. Yao, and J. Liu, "Overlooked video classification in weakly supervised video anomaly detection," in *WACV*, 2024, pp. 202–210.
[6] I. Bozcan and E. Kayacan, "Context-dependent anomaly detection for low altitude traffic surveillance," in *ICRA*. IEEE, 2021, pp. 224–230.
[7] S. Yu, Z. Zhao, H. Fang, A. Deng, H. Su, D. Wang, W. Gan, C. Lu, and W. Wu, "Regularity learning via explicit distribution modeling for skeletal video anomaly detection," *IEEE TCSVT*, vol. 34, no. 8, pp. 6661–6673, 2024.
[8] P. K. Mishra, A. Mihailidis, and S. S. Khan, "Skeletal video anomaly detection using deep learning: Survey, challenges, and future directions," *IEEE TETCI*, vol. 8, no. 2, pp. 1073–1085, 2024.
[9] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI*, vol. 32, no. 1, 2018.
[10] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *CVPR*, 2020, pp. 183–192.
[11] O. Hirschorn and S. Avidan, "Normalizing flows for human pose anomaly detection," in *ICCV*, 2023, pp. 13 545–13 554.
[12] C. Guo, H. Wang, Y. Xia, and G. Feng, "Learning appearance-motion synergy via memory-guided event prediction for video anomaly detection," *IEEE TCSVT*, vol. 34, no. 3, pp. 1519–1531, 2024.
[13] Y. Lu, C. Cao, Y. Zhang, and Y. Zhang, "Learnable locality-sensitive hashing for video anomaly detection," *IEEE TCSVT*, vol. 33, no. 2, pp. 963–976, 2023.
[14] R. Liang, Y. Li, J. Zhou, and X. Li, "Text-driven traffic anomaly detection with temporal high-frequency modeling in driving videos," *IEEE TCSVT*, pp. 1–1, 2024.
[15] Y. Zhong, X. Chen, Y. Hu, P. Tang, and F. Ren, "Bidirectional spatio-temporal feature learning with multiscale evaluation for video anomaly detection," *IEEE TCSVT*, vol. 32, no. 12, pp. 8285–8296, 2022.
[16] X. Zeng, Y. Jiang, W. Ding, H. Li, Y. Hao, and Z. Qiu, "A hierarchical spatio-temporal graph convolutional neural network for anomaly detection in videos," *IEEE TCSVT*, vol. 33, no. 1, pp. 200–212, 2023.
[17] Y. Zhao, B. Deng, C. Shen, Y. Liu, H. Lu, and X.-S. Hua, "Spatio-temporal autoencoder for video anomaly detection," in *ACM MM*, 2017, pp. 1933–1941.
[18] S. Yuan, L. Li, H. Chen, and X. Li, "Surface defect detection of highly reflective leather based on dual-mask guided deep learning model," *IEEE TIM*, 2023.
[19] W. Zhou, Y. Zhu, J. Lei, J. Wan, and L. Yu, "Ccafnet: Crossflow and cross-scale adaptive fusion network for detecting salient objects in rgb-d images," *IEEE TMM*, vol. 24, pp. 2192–2204, 2021.
[20] Y. Zhang, X. Nie, R. He, M. Chen, and Y. Yin, "Normality learning in multispace for video anomaly detection," *IEEE TCSVT*, vol. 31, no. 9, pp. 3694–3706, 2020.
[21] C. Chen, J. Wei, C. Peng, and H. Qin, "Depth-quality-aware salient object detection," *IEEE TIP*, vol. 30, pp. 2350–2363, 2021.

[22] M. Song, L. Li, D. Wu, W. Song, and C. Chen, "Rethinking object saliency ranking: A novel whole-flow processing paradigm," *IEEE TIP*, vol. 33, pp. 338–353, 2024.
[23] W. Zhou, Q. Guo, J. Lei, L. Yu, and J.-N. Hwang, "Ecffnet: Effective and consistent feature fusion network for rgb-t salient object detection," *IEEE TCSVT*, vol. 32, no. 3, pp. 1224–1235, 2021.
[24] J. Zhang, L. Qing, and J. Miao, "Temporal convolutional network with complementary inner bag loss for weakly supervised anomaly detection," in *ICIP*, 2019, pp. 4030–4034.
[25] H. Zhou, J. Yu, and W. Yang, "Dual memory units with uncertainty regulation for weakly supervised video anomaly detection," in *AAAI*, vol. 37, no. 3, 2023, pp. 3769–3777.
[26] P. Wu, J. Liu, X. He, Y. Peng, P. Wang, and Y. Zhang, "Toward video anomaly retrieval from video anomaly detection: New benchmarks and model," *IEEE TIP*, vol. 33, pp. 2213–2225, 2024.
[27] A. Acsintoae, A. Florescu, M.-I. Georgescu, T. Mare, P. Sumedrea, R. T. Ionescu, F. S. Khan, and M. Shah, "Ubnormal: New benchmark for supervised open-set video anomaly detection," in *CVPR*, 2022, pp. 20 143–20 153.
[28] Z. Yang, Y. Guo, J. Wang, D. Huang, X. Bao, and Y. Wang, "Towards video anomaly detection in the real world: A binarization embedded weakly-supervised network," *IEEE TCSVT*, vol. 34, no. 5, pp. 4135–4140, 2024.
[29] Y. Fan, Y. Yu, W. Lu, and Y. Han, "Weakly-supervised video anomaly detection with snippet anomalous attention," *IEEE TCSVT*, vol. 34, no. 7, pp. 5480–5492, 2024.
[30] Y. Zhou, Y. Qu, X. Xu, F. Shen, J. Song, and H. T. Shen, "Batchnorm-based weakly supervised video anomaly detection," *IEEE TCSVT*, pp. 1–1, 2024.
[31] A. Flaborea, L. Collorone, G. M. D. Di Melendugno, S. D'Arrigo, B. Prenkaj, and F. Galasso, "Multimodal motion conditioned diffusion model for skeleton-based video anomaly detection," in *ICCV*, 2023, pp. 10 318–10 329.
[32] A. Stergiou, B. De Weerdt, and N. Deligiannis, "Holistic representation learning for multitask trajectory anomaly detection," in *WACV*, 2024, pp. 6729–6739.
[33] W. Zhou, Y. Zhu, J. Lei, R. Yang, and L. Yu, "Lsnet: Lightweight spatial boosting network for detecting salient objects in rgb-thermal images," *IEEE TIP*, vol. 32, pp. 1329–1340, 2023.
[34] W. Zhou, F. Sun, Q. Jiang, R. Cong, and J.-N. Hwang, "Wavenet: Wavelet network with knowledge distillation for rgb-t salient object detection," *IEEE TIP*, vol. 32, pp. 3027–3039, 2023.
[35] W. Zhou, F. Sun, and W. Qiu, "Msnet: Multiple strategy network with bidirectional fusion for detecting salient objects in rgb-d images," *IEEE TASE*, 2024.
[36] W. Zhou, Q. Guo, J. Lei, L. Yu, and J.-N. Hwang, "Irfr-net: Interactive recursive feature-reshaping network for detecting salient objects in rgb-d images," *IEEE TNNLS*, 2021.
[37] H. Liu, L. He, M. Zhang, and F. Li, "Vadiffusion: Compressed domain information guided conditional diffusion for video anomaly detection," *IEEE TCSVT*, vol. 34, no. 9, pp. 8398–8411, 2024.
[38] L. Zanella, W. Menapace, M. Mancini, Y. Wang, and E. Ricci, "Harnessing large language models for training-free video anomaly detection," in *CVPR*, 2024, pp. 18 527–18 536.
[39] Y. Yang, K. Lee, B. Dariush, Y. Cao, and S.-Y. Lo, "Follow the rules: Reasoning for video anomaly detection with large language models," *arXiv preprint arXiv:2407.10299*, 2024.
[40] H. Paulheim, "Knowledge graph refinement: A survey of approaches and evaluation methods," *Semantic Web*, vol. 8, no. 3, pp. 489–508, 2017.
[41] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE TNNLS*, vol. 33, no. 2, pp. 494–514, 2021.
[42] P. Chandak, K. Huang, and M. Zitnik, "Building a knowledge graph to enable precision medicine," *Scientific Data*, vol. 10, no. 1, p. 67, 2023.
[43] F. Wang, Z. Zheng, Y. Zhang, Y. Li, K. Yang, and C. Zhu, "To see further: Knowledge graph-aware deep graph convolutional network for recommender systems," *Information Sciences*, vol. 647, p. 119465, 2023.
[44] G. Tamašauskaitė and P. Groth, "Defining a knowledge graph development process through a systematic review," *ACM TOSEM*, vol. 32, no. 1, pp. 1–40, 2023.
[45] G. Wang, Y. Wang, J. Qin, D. Zhang, X. Bao, and D. Huang, "Video anomaly detection by solving decoupled spatio-temporal jigsaw puzzles," in *ECCV*. Springer, 2022, pp. 494–511.
[46] J. Li, C. Wang, H. Zhu, Y. Mao, H.-S. Fang, and C. Lu, "Crowdpose: Efficient crowded scenes pose estimation and a new benchmark," in *CVPR*, 2019, pp. 10 863–10 872.
[47] M. Song, W. Song, G. Yang, and C. Chen, "Improving rgb-d salient object detection via modality-aware decoder," *IEEE TIP*, vol. 31, pp. 6124–6138, 2022.
[48] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *ICCV*, 2023, pp. 4015–4026.
[49] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, "Inpaint anything: Segment anything meets image inpainting," *arXiv preprint arXiv:2304.06790*, 2023.
[50] M. Sun, T. X. Han, M.-C. Liu, and A. Khodayari-Rostamabad, "Multiple instance learning convolutional neural networks for object recognition," in *ICPR*, 2016, pp. 3270–3275.
[51] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017, pp. 2980–2988.
[52] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.
[53] C. Cao, Y. Lu, P. Wang, and Y. Zhang, "A new comprehensive benchmark for semi-supervised video anomaly detection and anticipation," in *CVPR*, 2023, pp. 20 392–20 401.
[54] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection–a new baseline," in *CVPR*, 2018, pp. 6536–6545.
[55] S. Sun, J. Hua, J. Feng, D. Wei, B. Lai, and X. Gong, "Tdsd: Text-driven scene-decoupled weakly supervised video anomaly detection," in *ACM MM*, 2024, pp. 5055–5064.
[56] B. Degardin and H. Proença, "Human activity analysis: Iterative weak/self-supervised learning frameworks for detecting abnormal events," in *2020 IEEE IJCB*. IEEE, pp. 1–7.
[57] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.
[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[59] Y. Nie, H. Huang, C. Long, Q. Zhang, P. Maji, and H. Cai, "Interleaving one-class and weakly-supervised models with adaptive thresholding for unsupervised video anomaly detection," in *ECCV*. Springer, 2025, pp. 449–467.
[60] H. Lv, C. Chen, Z. Cui, C. Xu, Y. Li, and J. Yang, "Learning normal dynamics in videos with meta prototype network," in *CVPR*, 2021, pp. 15 425–15 434.
[61] M. Zhao, X. Zeng, Y. Liu, J. Liu, D. Li, X. Hu, and C. Pang, "Lgn-net: Local-global normality network for video anomaly detection," *arXiv preprint arXiv:2211.07454*, 2022.
[62] C. Cao, H. Zhang, Y. Lu, P. Wang, and Y. Zhang, "Scene-dependent prediction in latent space for video anomaly detection and anticipation," *IEEE TPAMI*, 2024.
[63] A. Markovitz, G. Sharir, I. Friedman, L. Zelnik-Manor, and S. Avidan, "Graph embedded pose clustering for anomaly detection," in *CVPR*, 2020, pp. 10 539–10 547.
[64] S. Sun and X. Gong, "Hierarchical semantic contrast for scene-aware video anomaly detection," in *CVPR*, 2023, pp. 22 846–22 856.

Chenglizhao Chen is a Professor in College of Computer Science and Technology, China University of Petroleum (East China). His research interests include virtual reality, computer vision, deep learning, data mining and pattern recognition.

Xinyu Liu is currently a M.S. student in the College of Computer Science and Technology, China University of Petroleum (East China). His research interests include computer vision and deep learning.

Mengke Song received the M.S. degree from Qingdao University in 2023. He is currently a Ph.D. student in China University of Petroleum (East China). His research interests include computer vision and deep learning.

Luming Li received the M.S. degree from Qingdao University in 2022. He is currently a Ph.D. student in China University of Petroleum (East China). His research interests include computer vision and deep learning.

Shaojiang Yuan received the M.S. degree from Wuhan Textile University in 2024. He is currently a Ph.D. student

in China University of Petroleum (East China). His research interests include computer vision and anomaly detection.

Xu Yu is a Professor in College of Computer Science and Technology, China University of Petroleum (East China). His research interests include recommended systems, intelligent software engineering and industrial Internet.

Shanchen Pang is a Professor in College of Computer Science and Technology, China University of Petroleum (East China). His research interests include software formalization, edge computing, artificial intelligence and computer vision.