



This is a repository copy of *AD-Graph: weakly supervised anomaly detection graph neural network*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/202397/>

Version: Published Version

Article:

Ullah, W. orcid.org/0000-0002-1243-9358, Hussain, T. orcid.org/0000-0003-4861-8347, Min Ullah, F.U. orcid.org/0000-0002-1243-9358 et al. (5 more authors) (2023) AD-Graph: weakly supervised anomaly detection graph neural network. *International Journal of Intelligent Systems*, 2023. 7868415. ISSN 0884-8173

<https://doi.org/10.1155/2023/7868415>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Research Article

AD-Graph: Weakly Supervised Anomaly Detection Graph Neural Network

Waseem Ullah ¹, Tanveer Hussain ², Fath U Min Ullah ³, Khan Muhammad ⁴,
Mahmoud Hassaballah ⁵, Joel J. P. C. Rodrigues ⁶, Sung Wook Baik ¹,
and Victor Hugo C. de Albuquerque ⁷

¹Sejong University, Seoul 143-747, Republic of Korea

²Institute for Transport Studies, University of Leeds, Leeds, UK

³Department of Electronic and Electrical Engineering, The University of Sheffield, Sheffield S10 2TN, South Yorkshire, UK

⁴Visual Analytics for Knowledge Laboratory (VIS2KNOW Lab), Department of Applied Artificial Intelligence, School of Convergence, College of Computing and Informatics, Sungkyunkwan University, Seoul 03063, Republic of Korea

⁵Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, AlKharj 16278, Saudi Arabia

⁶COPELABS, Lusófona University, Campo Grande 376, Lisbon 1749-024, Portugal

⁷Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza, Ceará, Brazil

Correspondence should be addressed to Khan Muhammad; khan.muhammad@ieee.org and Sung Wook Baik; sbaik3797p@gmail.com

Received 31 October 2022; Revised 1 June 2023; Accepted 14 June 2023; Published 31 July 2023

Academic Editor: Surya Prakash

Copyright © 2023 Waseem Ullah et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The main challenge faced by video-based real-world anomaly detection systems is the accurate learning of unusual events that are irregular, complicated, diverse, and heterogeneous in nature. Several techniques utilizing deep learning have been created to detect anomalies, yet their effectiveness on real-world data is often limited due to the insufficient incorporation of motion patterns. To address these problems and enhance the traditional functionality of anomaly detection systems for surveillance video data, we propose a weakly supervised graph neural-network-assisted video anomaly detection framework called AD-Graph. To identify temporal information from a series of frames, we extract 3D visual and motion features and represent these in a language-based knowledge graph format. Next, a robust clustering strategy is applied to group together meaningful neighbourhoods of the graph with similar vertices. Furthermore, spectral filters are applied to these graphs, and spectral graph theory is used to generate graph signals and detect anomalous events. Extensive experimental results over two challenging datasets, UCF-Crime and ShanghaiTech, show improvements of 0.35% and 0.78% against a state-of-the-art model.

1. Introduction

The rapid development of video surveillance systems and the underlying computer vision algorithms means that these play a vital role in monitoring human activities and preventing crime. These systems are implemented in smart cities to enable traffic monitoring, assist in law enforcement, and aid in an understanding of different anomalies. One task of video surveillance applications is anomaly detection; this is a very challenging problem, as unknown and possibly

abnormal events happen infrequently in real-world surveillance situations. The types of anomaly also vary with the type of application and the particular scene under surveillance. For instance, people running along a road with traffic may be defined as an anomaly, whereas people running on a football ground are considered having normal behaviour [1, 2].

Anomaly detection has a wide range of possible applications associated with ensuring public safety and security, preventing crime, and avoiding catastrophes, and one

essential aspect is therefore a real-time decision-making capability. For instance, events such as robbery, road accidents, and fires require immediate and automatic counteraction, which is made possible by the detection of the anomalous event in real time. In view of this, several anomaly detection algorithms have been introduced by researchers for use in this area. In the earlier stages of development, tracking using different trajectory-based techniques was applied to detect certain anomalies; for example, in [3], high-level intentionality features were extracted for use in intentional agent modelling, to move individuals and classify the trajectories of an intentional agent in a particle framework for anomaly detection. Deep-learning concepts have also been widely applied to tasks related to computer vision and have yielded excellent performance. The majority of these deep-learning methods are based on supervised learning (i.e., with labels); however, supervised or semi-supervised learning is also used in anomaly detection, requiring low training datasets [4–6]. Anomaly detection can be broadly divided into two categories, based on the deep-learning framework used: (i) frame generation and (ii) probability estimation. Methods based on probability estimation construct a model based on the features of the training set and calculate an anomaly score for the targets. For instance, the researchers in [7, 8] combined a parametric model with an autoencoder (AE) to estimate the probability distribution and detected anomalies via an autoregressive procedure. In [9], the authors combined an AE with a Gaussian mixture model to create an anomaly score. In contrast, methods based on frame generation produce one or more frames and authenticate them to detect anomalies.

In the past, a variety of handcrafted features have been extracted to handle problems related to computer vision and time series data. The main limitations of these methods involve the usage of traditional handcrafted feature engineering techniques, and data-driven approaches are more favourable in the later stages. The emergence of deep-learning (DL) methods has solved a wide range of problems faced by conventional techniques. For instance, Chong and Tay [10] proposed a DL model comprising a recurrent neural network (RNN) and convolutional filters. These approaches are able to learn long-term contextual dynamics; i.e., the motion and the appearance are encoded implicitly by these methods within the proposed neural model. Although these approaches have shown good performance, they suffer from two limitations: First, the motion and appearance are encoded using an RNN and convolutional filters, meaning that the spatiotemporal relationship between the motion and appearance is missing, which yields inferior performance. Second, the features are learned from scratch without a well-developed pretrained model. Handling complex anomalies with such approaches becomes difficult when applied to real-world surveillance videos.

Nowadays, convolutional neural networks (CNNs) are applied for numerous computer vision undertakings including activity recognition [11–13] and summary analysis [14, 15]. However, these networks often encounter difficulties when applied to complex scenes for the detection of anomalous events. Approaches based on probability

estimation can generate more flexible frameworks for detecting anomalies by accommodating normal scenes at a spatiotemporal scale; however, these methods rely on probability models, which have difficulty in simulating the complicated distributions of different events and lead to the generalisation of an unobserved event while reducing the sensitivity to unfamiliar anomalies. Similarly, existing anomaly detection methods have problems when faced with occlusion or illumination issues and are mostly based on traditional ways of detecting anomalous events in surveillance scenarios.

The identification of video anomalies using mainstream methods often requires complex DL architectures with a large number of parameters, which poses a computational challenge. High-resolution surveillance videos are particularly susceptible to this problem. In order to achieve a high level of accuracy with acceptable computational efficiency, efficient architectures must be developed. In order to handle these challenges and problems, we propose an efficient graph neural network (GNN)-based anomaly detection method. The proposed model constructs a graph in which cars or people are represented as nodes and analyses their movements in surveillance videos. When applied to a parking lot, for instance, the proposed model can identify unusual activities such as cars driving against the traffic flow or lingering in one spot for extended periods, which may indicate suspicious behaviour. Similarly, in a crowded shopping mall, the model can detect potential security threats such as groups of people gathering suspiciously or an individual loitering in one area for too long. The data from the graph nodes are used in the proposed model to improve the accuracy of anomaly detection and enhance security and safety in real-world applications.

An overview of the system is shown in Figure 1, which illustrates how the detection of anomalous surveillance events is improved through better knowledge of the connections between graph nodes. The segments of the surveillance video are first passed through the backbone model, and meaningful visual features are extracted. From these features, language-based knowledge graphs are generated that are passed from the GNN. In addition to updating the graph, the output of the GNN is used to detect instances of anomalies and normal events from the graph, as well as the losses between the nodes. We propose a straightforward method for updating the graph's structure which involves learning better node representations and using them to recompute the adjacency matrix. The contributions of our approach can be summarised as follows:

- (1) We propose a mechanism for anomaly detection in surveillance systems in which a GNN is trained in a weakly supervised manner. This method incorporates both attributes and graph structure information. A language-based knowledge graph is generated using time series motion and appearance similarities to represent the conceptual relationships within the video sequence.
- (2) Mainstream anomaly detection approaches rely on 2D motion or appearance information to handle

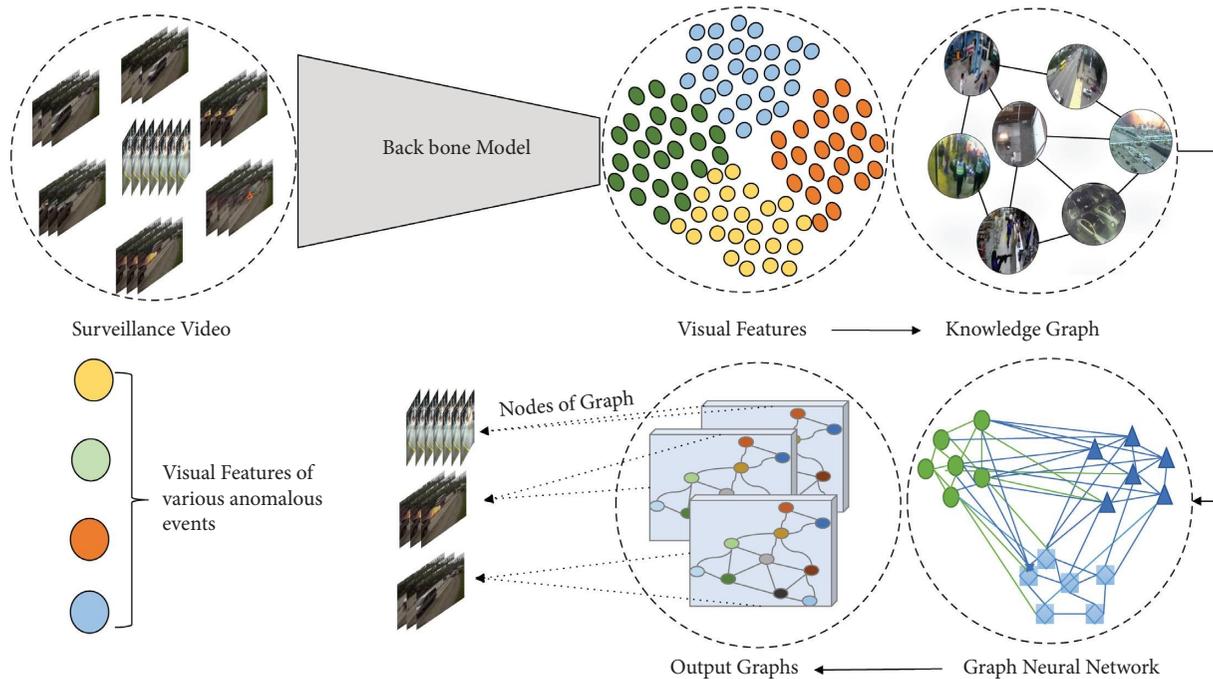


FIGURE 1: Overview of the proposed AD-Graph system, in which knowledge graphs are used for video anomaly detection. Each knowledge graph is based on various unusual activities and visual features extracted from surveillance videos showing each anomalous event.

problems such as complex backgrounds and variable illumination in surveillance videos and overlook the most important details hidden in sequential patterns. To tackle this challenge, we employ 3D visual and motion features to interconnect similar frame segments in the graph through the computation of frame-level characteristics, which enables precise anomaly detection.

- (3) The majority of the graphs produced in this way are highly complex and dynamic, making detection more computationally demanding. We therefore apply a clustering strategy to group similar vertices, whereby the vertices are rearranged and a balanced binary tree is constructed from the clustered graph. To form a graphed signal, a pooling operation is applied, and a regular 1D signal is generated that leads towards the optimal output.
- (4) We experimentally show that the proposed method efficiently uses the nodes of the graph to detect anomalies and outperforms state-of-the-art (SOTA) models on publicly available benchmarks.

The paper consists of four sections. In Section 2, we present a literature review that contextualises this research and identifies gaps in the field of anomaly detection. Section 3 introduces the proposed AD-Graph method with a detailed description of its design, architecture, and capabilities. In Section 4, we discuss the experiments conducted in the study, including the methodology used, the results obtained, and their implications. Finally, in Section 5, we summarise our findings, identify their potential impacts, and suggest areas for future research.

2. Related Work

The domain of anomaly detection is diverse and involves a wide variety of settings and assumptions, as is obvious from the numerous datasets that have been created to assess the existing algorithms in this field. The identification of anomalies from video data has been the subject of several studies in recent years. These have mainly relied on DL-based techniques, which can be divided into three categories: (i) unsupervised methods, (ii) weakly supervised methods, and (iii) temporal dependency-based anomaly detection techniques, as discussed below.

2.1. Unsupervised Anomaly Detection. Unsupervised learning is often the approach taken for anomaly detection when the training phase does not include abnormal events. Traditional approaches undertake the accessibility of ordinary training samples and handle anomaly detection as a one-class classification problem via traditional features. Thanks to recent rapid advancements in DL, modern approaches now adopt features from pretrained deep neural networks [16–18]. Alternatively, to learn compact normality representation constraints in the latent space, a normal manifold strategy can be applied [19–21], where any diversion or small change from the normal patterns in the same latent space is considered to be an abnormal event. In addition to these approaches, data reconstruction techniques can be used to acquire information of usual events by reducing the reconstruction error using generative models [22–24]. Spectral-based (also called subspace-based) techniques and neural network approaches are types of reconstruction-based methods. When anomalies are projected into

a lower-dimensional space, these approaches presume that information is lost and cannot be successfully recreated. However, these methods show limited performance in terms of distinguishing between normal and abnormal events because of a deficiency of prior knowledge about abnormal events.

2.2. Weakly Supervised Anomaly Detection (WSAD). Significant improvements have been observed in the performance of unsupervised approaches with the partial leverage of labelled samples [12, 25, 26]. The creation of labelled annotations for each frame in large amounts of surveillance data is an expensive task. SOTA approaches therefore apply weakly supervised training by adopting cheap annotation methods for anomaly detection [25–29]. In one existing study [25], the authors described the usage of video-level annotations and established a challenging WSAD dataset called UCF-Crime. The WSAD from videos has been highlighted as a research area of intense interest [29–31]. These approaches are purely established on the use of multiple instance learning methods [25], whereas numerous methods based on multiple instance learning are not effective to leverage abnormal annotations, as they are affected by the noisy annotations from the positive bag produced by a normal segment that is inaccurately chosen as the top irregular event in an anomaly video. In one prior study [27], the authors handled this issue as a binary class classification problem with noisy annotation and adopted a graph convolution neural network (GCNN) to clear all the noisy annotations. Training a GCNN and multiple instance learning at the same time is computationally very expensive and also leads to an unconstrained latent space, which causes unstable performance. Despite these limitations, however, this method has shown more accurate results than the multiinstance learning approach [25].

2.3. Temporal Dependencies. Various traditional approaches to anomaly detection have explored the use of temporal dependencies [22, 27]. For instance, in [32], the authors converted sequences of video frames into handcrafted motion features to highlight the regional consistency among adjacent frames. In DL-based anomaly detection, diverse types of sequential information have been adopted, for example, to predict sequential consistency in upcoming video frames [22], with a stacked RNN [33], a dual stream approach [34, 35], or convolutional long short-term memory [36]. Some of these approaches are based on a small fixed-ranged temporal correlation; for instance, five frames per sequence are considered in a stacked RNN [33]. Others employ the long-range dependencies of all possible sequential locations and events with variable sequence lengths. Vision transformer-based models have also been used in recent studies [37, 38]; these techniques use a transformer for feature extraction and sequential pattern learning. The authors of [37] proposed a hybrid technique called TransCNN, in which a CNN was applied to extract features from an input video and a transformer mechanism was then used to acquire the temporal relationships between these

features. Another baseline method has been proposed for surveillance anomaly recognition models. The authors of [38] used a vision transformer with a multihead attention mechanism for feature extraction, and a multireservoir with an attention model was designed for temporal pattern learning. In some recent studies [27, 39], the authors have explored GCN-based approaches to collect the long-range dependencies among segmented features. In contrast, our AD-Graph technique is based on motion and visual appearance and establishes relationship graphs of the similarities between low- and high-confidence snippets. Spectral representation of these snippets is applied to process the graph signal for accurate classification of anomalous and normal events.

3. The Proposed AD-Graph

Our objective in this work was to develop a temporal anomaly detection system, as the temporal interactions between objects are very informative for use in predicting anomalous events or incidents from a video. During the training phase, we utilize videos with weak labels, wherein we are able to identify the events occurring in the video but lack information regarding the timing and frequency of anomalies. In order to train our model, we use weakly labelled videos. The proposed method is based on three steps: (i) extraction of temporal features, (ii) a graph coarsening process that combines related edges, and (iii) a graph grouping process that transfers higher filter resolution with spatial resolution.

3.1. Architecture of AD-Graph. The design of our AD-Graph is depicted in Figure 2. It showcases the input, denoted as l which represents a volume of features. In this context, l refers to the number of time segments in the video, while d_{in} represents the dimension of the features. Each time segment is defined as x_i , and the total number of input features is \mathcal{X}_i . The input 3D features are then modified by applying a graph convolution layer. We employ optical flow and RGB-based similarity with the graph's weight edges, in which a distinct linear layer μ is the similarity metric. For each input time segment, AD-Graph generates a prediction score for every class. The final prediction score, denoted as l/c volume, corresponds to the value Y . Here, c represents the total number of anomaly detection classes.

3.2. Feature Extraction. The aim at this stage is to extract the prime representative features from the video. To do this, we leverage a frame-level feature extraction inception model that has been previously trained on the Kinetics dataset [41]. In order to show every video segment, we extract 3D features from the videos following the method in [42]. Specifically, every video consists of two sets l of 1024 volumes features. The first set originates from an RGB-based stream, while the second set comes from an optical flow-based stream. Here, l represents the count of time segments used as input. These dual features are combined to present a final feature vector of

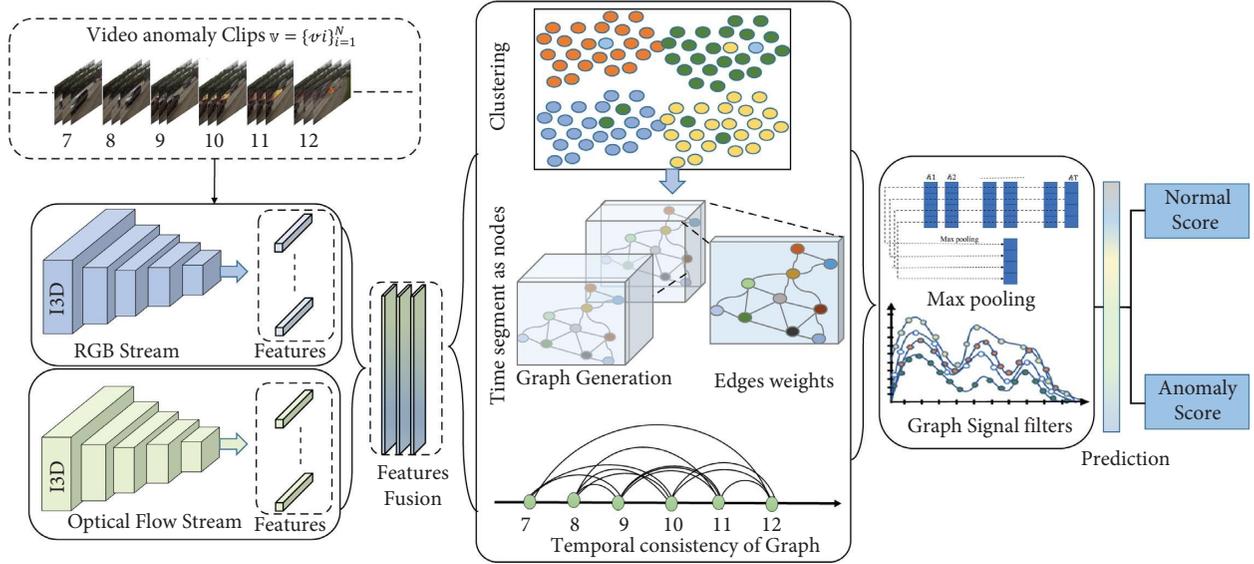


FIGURE 2: Pipeline used in the proposed AD-Graph system for video anomaly detection. 3D features are extracted from video snippets that are grouped together using the Graclus multilevel clustering algorithm. A graph is generated to model the temporal consistency and the similarity among these snippet features. A Laplacian filter is used to achieve Chebyshev decomposition [40], and max pooling is applied prior to the final prediction, in which graph signals are used to predict the event as anomalous or normal.

size $l \times 2048$. The notation utilized in this work is listed in Table 1.

3.3. Graph Construction. GNNs and graph convolutional networks (GCNs) have attracted considerable interest in various domains such as action recognition [43, 44], video summarisation [45, 46], surveillance [27, 39], and healthcare [47, 48]. In a GCN, the input time is divided into segments, and each segment is considered a node in a graph. Interpretation is then carried out over this graph. Its similarity is measured on the node edges, enabling similar time chunks to be combined and irrelevant time segments to be separated in the feature space, thereby informing each other simultaneously during the train and test stages. During this stage, graph convolutions can help improve localisation by requiring the network to analyse and evaluate each sequence class against the background of other segments of the duration, which are both similar and distinct. The following transformation is performed by the graph layer on \mathcal{X} :

$$\mathcal{Z} = \mathcal{G} \mathcal{X} \mathcal{W}, \quad (1)$$

where \mathcal{W} is weighted matrix of size $2048 \times d_{\text{out}}$ acquired using backpropagation, \mathcal{Z} is the output of graph with size $l \times d_{\text{out}}$, and \mathcal{G} is a normalised affinity matrix. This matrix \mathcal{G} has size $l \times l$, and \mathcal{G}_{ij} are the edge weights for x_i and x_j .

Our aim is to process the defined signals on connected and undirected graphs $\mathcal{g} = (\mathcal{E}, \nu, \omega)$, where \mathcal{E} is the edge, ν is the set of vertices, and $\omega \in \mathbb{R}^{n \times n}$ is the adjacent matrix encoding of the weights for two consecutive vertices. The signal of $x: \nu \rightarrow \mathbb{R}$ graph connections is considered a vector $x \in \mathbb{R}^n$. The Laplacian graph is a vital operator that is used to perform spectral analysis [49] and is expressed as $\mathcal{L} = D - \omega \in \mathbb{R}^{n \times n}$. The diagonal degree matrix is represented as $D \in$

TABLE 1: Notation used in AD-Graph.

Notations	Meaning
l	Volume of features
d_{in}	Dimensions of features
x_i	Individual segments
\mathcal{X}_i	Total number of input features
M	Similarity metric
\mathcal{G}	Normalised affinity matrix
\mathcal{W}	Weighted matrix
\mathcal{g}	Undirected graphs
\mathcal{E}	Edges
ν	Vertices
Y	Final prediction

$\mathbb{R}^{n \times n}$; each element along the diagonal is $D_{i=i} = \sum_j \omega_{ij}$, which in normalised form is calculated as $\mathcal{L} = \mathbf{I}_n - D^{-1/2} \omega D^{-1/2}$, where \mathbf{I}_n represents the identity matrix. Taking into consideration the eigen decomposition of \mathcal{L} as $\mathbf{U} \Lambda \mathbf{U}'$, Λ is the matrix called as graph Fourier modes (eigen vectors), and diagonal elements of the matrix being non-negative eigen values are the frequencies of the graph [50]. Spectral convolution is a distinct graph operator that is obtained by primarily projecting a provided graph signal and applying the eigen decomposition of \mathcal{L} , and before the backproject, we multiply the resulting projections in the original signal space by a convolution filter. The convolution filters \mathcal{G} applied to the graph signal are defined as $\psi(\nu) \in \mathbb{R}^{n \times n}$, where $(\psi * \mathcal{G}\theta)(\nu) = \mathbf{U} \mathcal{G}\theta(\Lambda) \mathbf{U}' \psi(\nu)$. $\mathcal{G}\theta$ represents those filters for which all parameters are free, also known as non-parametric filters; these filters are expressed as $\mathcal{G}\theta(\Lambda) = \text{diag}(\theta)$, where the parameter $\theta \in \mathbb{R}^n$ is a Fourier coefficient vector. The main problem with nonparametric filters is that they cannot be localised in space dimensionality of data, and we therefore use the following alternative [49]:

$$(\psi * \mathcal{L}\theta)(\nu) = \sum_{K=0}^{\kappa-1} \theta_K \mathcal{C}_K(\mathcal{L})\psi(\nu). \quad (2)$$

In equation (2), κ is constant and $\theta \in \mathbb{R}^\kappa$ are the learned parameters of the convolutional filter. We apply a normalised Laplacian variant at the training stage, i.e., $2\mathcal{L}/\lambda_{\max}\mathbf{I}_n$ as an alternative \mathcal{L} , including λ_{\max} as the highest eigenvalue. \mathcal{C}_K is the Chebyshev k th order polynomial, recursively expressed as $\mathcal{C}_K(\mathcal{L}) = 2\mathcal{L}\mathcal{C}_{K-1}(\mathcal{L})$, where $\mathcal{C}_K(\mathcal{L}) \in \mathbb{R}^{n \times n}$ and $\mathcal{C}_0 = \mathbf{I}, \mathcal{C}_1 = \mathcal{L}$. For further details, the reader is referred to the explanation given in [50]. A list of abbreviations used in the AD-Graph model is presented in Table 2.

3.4. Graph Coarsening and Pooling. For the pooling process, significant neighbourhoods are necessary in graphs where identical vertices are clustered. This process is similar to multilayer graph clustering, which retains geometric local structures. However, this graph clustering problem is NP-hard [51], meaning that approximations must be applied. Although there are several possible clustering approaches, such as the well-known spectral clustering method [52], we are particularly interested in a multilevel clustering method with a coarser graph for each level that corresponds to various data domains. In addition, a clustering method in which the graph size is reduced by a factor of two at each level provides accurate control of pooling size and coarsening. In this research, the Graclus multilevel clustering method is used, as it has been proved to be highly effective for clustering large numbers of graphs. However, this method of graph coarsening generally leads to unbalanced hierarchical representations for extremely irregular graphs, which significantly influences the precision of the acquired graph presentations. Pooling operations are performed several times and must be effective for multilevel graph coarsening. The coarsened graph and the vertices of the input graph are not organised in any meaningful way after coarsening. Thus, a table would be required to store all the matched vertices to allow the pooling process to be applied directly. This would lead to inefficient memory usage, a lack of parallelisation, and slow implementation. However, the vertices may be arranged to make the graph pooling process as efficient as 1D pooling. In this paper, we explore an alternate approach to pooling. Our technique involves two stages: an expansion process is first performed at the node level, and average global pooling is then applied to ensure permutation invariance [50]. The first stage is required in order to build large (and scant) node representations and thereby retain the discriminating power of the nodes before global average pooling is carried out in the second stage. In other words, average pooling without expansion enables permutation invariance but dilutes the node information and leads to less discriminating graphs, as demonstrated in tests. The main objective of this work is to model temporal visual and motion similarity relationships among the time segments of a surveillance video to detect anomalous events. For this purpose, we use a GNN to treat the given features as nodes in a graph, and a clustering algorithm is

TABLE 2: Abbreviations used for surveillance anomaly detection.

Abbreviations	Description
CNN	Convolutional neural network
AE	Autoencoder
RNN	Recurrent neural network
GNN	Graph neural network
2D	Two-dimensional
3D	Three-dimensional
GCNN	Graph convolution neural network
AD-graph	Anomaly detection graph
I3D	Inflated 3D networks
AUC	Area under the curve

then employed to make use of coarsening to efficiently cluster the huge variety of graphs. Graph Laplacian filters are applied to the weighted edges to convert the graph signal, for precise classification of normal and anomalous events. There are several key advantages to using the proposed model based on GNN to detect anomalous events from surveillance videos. For instance, it provides an effective way to model the temporal relationships between visual and motion features over different time segments, which are crucial for detecting anomalies that may develop over time. In addition, graph clustering is used to represent complex and varied graphs in an efficient way, which is important for handling both large and diverse input datasets. Furthermore, the application of graph Laplacian filters to the edge weights enables accurate differentiation between normal and anomalous events. As a result, the proposed model offers a powerful approach for detecting anomalous events in surveillance videos and has important applications in the fields of security and safety.

4. Results and Discussion

The proposed AD-Graph was evaluated on various challenging video anomaly detection datasets. The performance comparison was carried out using unsupervised and WSAD techniques to test the effectiveness of our AD-Graph against SOTA alternatives, and our method was found to give the best performance. We compared the performance of the proposed AD-Graph with 22 other recent models using both supervised and unsupervised techniques, as summarised in Table 3. We also present quantitative and qualitative results from the proposed AD-Graph method that highlight the improvements and achieves over SOTA techniques which do not explicitly model the connections among time segments.

4.1. Dataset and Evaluation Metrics. In this work, we use two challenging recent datasets to evaluate our AD-Graph model, called UCF-Crime and ShanghaiTech, which have predefined training and test sets. To ensure a fair evaluation of AD-Graph, we use the standard evaluation protocol used in the prior studies [22, 25, 29, 31], based on the ROC curve and the frame-level area under the curve (AUC) for all datasets.

TABLE 3: AUC performance of AD-Graph compared with other weakly supervised and unsupervised techniques on the UCF-Crime and ShanghaiTech datasets.

Methods	Years	Backbone	ShanghaiTech AUC (%)	UCF-Crime AUC (%)
Hasan et al. [53]	2016	Fully convolutional AE	60.90	50.60
Luo et al. [33]	2017	Stacked RNN	68.00	—
Liu et al. [54]	2018	Ensemble classifier strategies	72.80	—
Sohrab et al. [55]	2018	C3D	—	58.50
Sultani et al. [25]	2018	C3D	—	75.41
Gong et al. [56]	2019	Memory-based AE	71.20	—
GODS [57]	2019	One class learning	—	70.46
Zhu and Newsam [26]	2019	Temporal augmented network	—	79.00
GCN-Anomaly [27]	2019	I3D	—	82.12
Dong et al. [58]	2020	Dual GAN	73.70	—
Doshi et al. [59]	2020	CNN-KNN	71.60	—
Park et al. [20]	2020	Memory module	70.50	—
Tang et al. [60]	2020	U-Net generator	73.00	—
Chang et al. [61]	2020	Clustering-based deep AE	73.30	—
Ano-Graph [39]	2021	Spatial temporal graph	74.42	—
Tangqing et al. [62]	2021	Clustering-based AE	—	72.90
Doshi et al. [63]	2021	Hybrid modules	70.90	—
Chandrakala et al. [64]	2022	Bag of events model	—	83.50
Liu et al. [65]	2022	Dual stream AE	73.60	—
EDM [66]	2023	Diffusion model	—	65.22
STR-VAD [67]	2023	Spatial temporal	73.2	—
RAE [68]	2023	Residual AE	73.60	—
AD-Graph (ours)		RGB-optical flow (I3D)	75.20	83.85

The best results are highlighted as bold text in the table.

4.1.1. ShanghaiTech. This is a medium-sized dataset of static-angle street surveillance videos, with 13 different background scenes. It contains a total of 437 videos, of which 130 contain anomalous events and 307 normal scenes. This dataset is a well-known benchmark for anomaly detection, in which the training data include only normal videos [22]. The dataset was reorganised by Zhong et al. [27] to create a weakly supervised video labelled by choosing a subset from the anomalous event test set into the training set to cover 13 background events in both the training and testing sets. We performed experiments on this weakly supervised dataset, as described in [27, 30, 31].

4.1.2. UCF-Crime. UCF-Crime is a large-scale dataset of anomalous events that contains 1,900 long and untrimmed videos [25]. The total duration of these videos is 128 h, and they include 13 types of anomaly recorded in real-world indoor and outdoor surveillance environments. Unlike the stationary backgrounds contained in ShanghaiTech, these videos have a diverse range of complicated backgrounds. This is a relatively balanced dataset that contains equal numbers of normal and unusual events in both the training and test sets. The challenging aspect of this dataset is the lack of temporal annotation for the training videos, with only video-level labels and test videos.

4.2. Implementation Details. To describe the proposed AD-Graph architecture, we use the following notation: F_{C_k} , p_k , C_k , and GC_k are the fully connected, pooling, and convolutional layers with k hidden units, stride and size, and feature maps, respectively. The ReLU activation function is

used for GC_k , p_k , and C_k . The output of GC_k is passed to the nonlinear ReLU activation function, normalised with ℓ_2 regulation, and then input to the linear Softmax classification layer with a batch size of 128. A dropout rate of 0.5% is used in the graph and linear layers. We train AD-Graph for 120 epochs with the Adam optimiser and a learning rate of 0.001. We build \mathcal{G} during both training and testing from video time segments at a time. During both the training and testing processes, AD-Graph constructs a graph from video segments and then processes each segment individually. This allows the model to analyse the temporal relationships between the frames within each segment and to identify any anomalies present in the video data. The process of extracting features from video frames in AD-Graph follows the baseline method in [27]. More specifically, features are extracted from clips containing 16 frames. To ensure consistency, each video is divided into T snippets, and the average is computed for all 16-frame clip-level features within each snippet. This approach helps maintain a consistent number of snippets in each video and allows for effective feature extraction from the video data. The implementation was based on Python 3.6 and TensorFlow, and the AD-Graph model was tested on a GeForce Titan-X graphics processing unit.

4.3. Experimental Evaluation of AD-Graph on ShanghaiTech. The experimental results on the ShanghaiTech dataset using frame-level AUC are shown in Table 4. A comprehensive evaluation of the AD-Graph network reveals that it achieves superior results against weakly supervised SOTA techniques. AD-Graph achieves AUC results that are 75.2% better than the weakly supervised-based approaches in [39, 61], and

TABLE 4: Performance comparison of AD-Graph with other existing weakly supervised and unsupervised techniques on the ShanghaiTech dataset

Methods	Years	Approach	AUC (%)
Hasan et al. [53]	2016	Unsupervised	60.85
Luo et al. [33]	2017		68.00
Liu et al. [54]	2018		72.80
Gong [56]	2019		71.20
Doshi and Yilmaz [63]	2021		70.90
Liu et al. [65]	2022		73.60
RAE [68]	2023		73.60
Dong et al. [58]	2020		Weakly supervised
Doshi and Yilmaz [59]	2020	71.60	
Park et al. [20]	2020	70.50	
Tang et al. [60]	2020	73.00	
Chang et al. [61]	2020	73.30	
Ano-Graph [39]	2021	74.42	
STR-VAD [67]	2023	73.2	
AD-Graph (ours)		75.20	

[60], with increases in the AUC scores of 0.78%, 1.9%, and 2.2%, respectively. In terms of accuracy, AD-Graph outperforms Ano-Graph [39] and the methods put forward by Chang et al. [61], Yang et al. [60], Dong et al. [58], and Park et al. [20] by 0.78%, 1.9%, 2.2%, 1.5%, and 4.7%, respectively. In addition, our AD-Graph outperforms the unsupervised methods in [56, 63, 65] with increases in 1.6%, 4.3%, and 4%, respectively.

4.4. Experimental Evaluation of AD-Graph on UCF-Crime. The AUC results for UCF-Crime are displayed in Table 5. AD-Graph achieved better results than the weakly supervised schemes in [25–27, 62, 64] and the unsupervised techniques in [53, 55, 57]. In terms of accuracy, it outperformed the recent unsupervised SOTA GODS method [57] with an increase of 12.39%. It was also interesting to observe that AD-Graph outperformed the weakly supervised-based SOTA methods, i.e., those of Hasan et al. [53], Sohrab et al. [55], Sultani et al. [25], Yi Zhu and Shawn Newsam [26], GCN-Anomaly [27], Li et al. [62], and Chandrakala et al. [64], by 32.25%, 24.35%, 12.39%, 3.85%, 0.73%, 9.6%, and 0.65%, respectively. The accuracy of AD-Graph was also compared with unsupervised approaches [53, 55, 57] and was found to be superior, with increases of 32.25%, 24.35%, and 12.39%, respectively.

4.5. Anomaly Heterogeneity. The UCF-Crime dataset contains various anomalous events, such as stealing, shooting, and road accidents. We therefore analysed the ability of our AD-Graph model to distinguish anomalous events from normal events, as shown in Figure 3, and found that it was superior to the baseline approach [25]. To explore the AUC performance for each individual anomaly class, we performed an experiment on the UCF-Crime dataset. To train the model, we used the full training and testing datasets from UCF-Crime and considered the scheme in [25] as a baseline to test the performance of our method. AD-Graph showed

TABLE 5: Comparison of AUC results for AD-Graph with other existing weakly supervised and unsupervised techniques on the UCF-Crime dataset.

Methods	Years	Approach	AUC (%)
Hasan et al. [53]	2016	Unsupervised	50.60
Sohrab et al. [55]	2018		58.50
GODS [57]	2019		70.46
EDM [66]	2023		65.22
Sultani et al. [25]	2018	Weakly supervised	75.41
Yi Zhu and Shawn Newsam [26]	2019		79.00
GCN-Anomaly [27]	2019		82.12
Li et al. [62]	2021		72.90
Chandrakala et al. [64]	2022		83.50
BSPR [69]	2022		83.39
AD-Graph (ours)			83.85

The best results are highlighted as bold text in the table.

good performance and outperformed SOTA when compared with the outcomes from the baseline techniques for individual anomaly classes. The results are shown in Figure 3 and confirm the superiority of our AD-Graph approach for almost every class, even for abnormalities that are very subtle. Our AD-Graph surpassed the baseline model [25] in 11 classes of anomalous events (abuse, arrests, fights, shootings, arson, shoplifting, road accidents, assaults, theft, explosions, and vandalism), with a noticeably improved AUC performance of 5% to 17% for the most confusing classes. For burglary and robbery, AD-Graph was less effective, but its performance was competitive with that of the baseline [25].

4.6. Ablation Study of Various 3D Models. The statistics in Table 6 reveal the considerable advantages of the proposed AD-Graph method in terms of its anomaly detection performance compared to recent 3D models. Primarily, we selected the I3D model [41] for experiments on the UCF-Crime dataset, in which we considered I3D-RGB features, I3D optical-flow-based features, and a combination of I3D RGB-optical flow features, followed by experiments on a C3D model [70]. We also tested other 2D CNN models such as ResNet10, InceptionV3, ResNet152, and VGG19. The results of our ablation study show that ResNet101 with RGB modality achieved the highest AUC score of 65.30% among the tested 2D CNN backbones. InceptionV3 with RGB modality achieved an AUC score of 64.70%, which was slightly lower than that of ResNet101. The use of ResNet152 or VGG19 with RGB modality resulted in even lower AUC scores of 61.30% and 58.50%, respectively. ResNet101 with RGB modality achieved the highest AUC score of 65.30% among the tested 2D CNN backbones, but this performance was still lower than that of the I3D model with RGB-optical flow modality. The best results were observed for AD-Graph with RGB-optical flow; the reason for this is that this method considers both motion and appearance features to detect anomalous acts, thereby providing good detection performance.

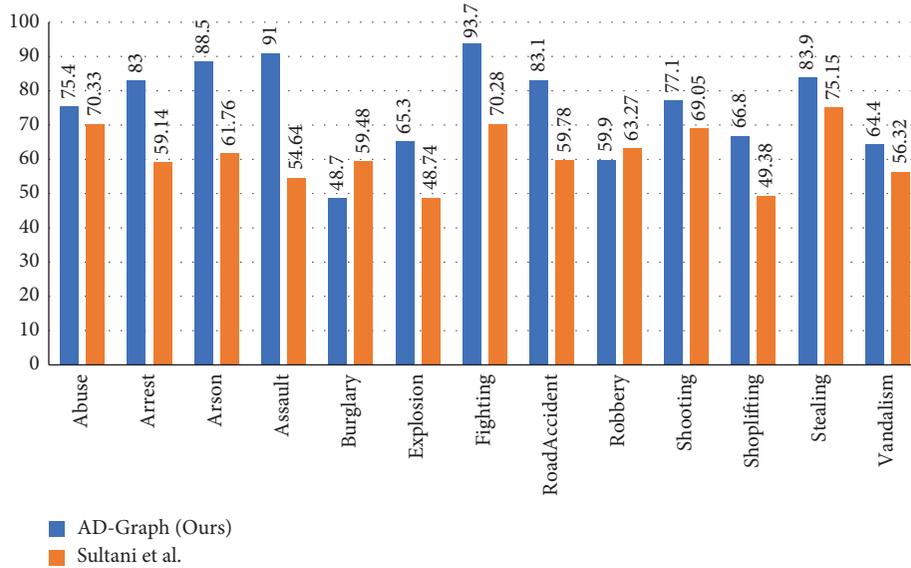


FIGURE 3: Performance comparison between AD-Graph and the method of Sultani et al. [25] in terms of AUC results for the classes of anomalies in the UCF-Crime dataset.

TABLE 6: Ablation study of various 3D models with AD-Graph.

Backbone	Modality	AUC (%)
I3D [41]	RGB	78.62
I3D [41]	Optical flow	80.14
I3D [41]	RGB-optical flow	81.56
ResNet101		65.30
InceptionV3		64.70
ResNet152	RGB	61.30
VGG19		58.50

4.7. Comparison and Discussion. To evaluate the performance of our AD-Graph technique, we compared it with existing alternatives using the ShanghaiTech and UCF-Crime datasets. We used the test sets of these datasets to determine the AUC and compared our results with those of 19 SOTA models, including both supervised and unsupervised techniques, as presented in Table 3. The AUC values in Table 3 show that the proposed AD-Graph model achieved the highest score of 83.85%, while the model of Chandrakala et al. [64] achieved the next highest score of 83.50%. The schemes presented in [25–27, 53, 55, 57, 62] achieved AUC values of 75.41%, 79.00%, 82.12%, 50.6%, 72.90% 58.50%, and 70.46%, respectively, on the UCF-Crime dataset.

Our proposed AD-Graph method also achieved the highest AUC performance on the ShanghaiTech dataset with a value of 75.2%, representing an increase of 0.78% compared with the recent Ano-Graph approach [39]. The techniques described in [20, 33, 53, 54, 56, 58, 60, 61, 63, 65] achieved AUC scores of 70.5%, 68.0%, 73.7%, 73.0%, 73.3%, 60.90%, 72.8%, 71.2%, 73.6%, and 70.9%, respectively. These results for AD-Graph indicate that it has the potential to improve the accuracy of anomaly detection in surveillance videos and could be used in real-world applications to enhance security and safety. However, it is important to note

that the performance of AD-Graph may vary across different datasets and scenarios, and further research is needed to evaluate its robustness and generalizability. Overall, our findings suggest that AD-Graph is a promising approach for detecting anomalies in surveillance videos and can contribute to the development of more effective and reliable surveillance systems.

5. Conclusion

In the field of video anomaly detection, learning-based systems are used to detect abnormal behaviour from video streams. However, the design of effective deep-learning solutions is difficult due to the low interpretability of the models. In this paper, we propose a new mechanism for anomaly detection based on a weakly supervised method called AD-Graph. The main strength of our technique is its ability to train multiple Laplacian convolutional operators, each of which is assigned to a certain configuration of the manifold comprising the input graph data.

The primary conclusions that could be drawn from this work were as follows. In general, the learning of short- and long-term temporal relations is vital for anomaly detection when training an end-to-end model. To learn the temporal relationships among video segments, we extracted 3D features from the input video and generated graphs based on the similarity between these segments. Our approach improved the AUC performance for anomaly detection by 1.5% and 0.73% on the ShanghaiTech and UCF-Crime datasets, compared with SOTA techniques. Experimental results for these challenging datasets indicated that AD-Graph achieved considerably better performance than existing weakly supervised and unsupervised video anomaly detection techniques, thus proving the effectiveness of our approach. Our model experiences difficulties in certain

challenging situations, such as images with low resolution, low levels of illumination, fast motion, and groups of people. In future work, we aim to solve these problems using incremental learning, explainable AI, and temporal transformation-based self-supervision.

Data Availability

The datasets that has been used in this work is publicly available.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT), Grant/award number: (2023R1A2C1005788).

References

- [1] S. Tariq, S. Tariq, H. Farooq, A. Jaleel, and S. M. Wasif, "Anomaly detection with particle filtering for online video surveillance," *IEEE Access*, vol. 9, pp. 19457–19468, 2021.
- [2] H. Sun, Q. He, and K. Liao, "Fast anomaly detection in multiple multidimensional data streams," in *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*, pp. 1218–1223, Los Angeles, CA, USA, December 2019.
- [3] F. Tung, J. S. Zelek, and D. A. Clausi, "Goal-based trajectory analysis for unusual behaviour detection in intelligent surveillance," *Image and Vision Computing*, vol. 29, no. 4, pp. 230–240, 2011.
- [4] D. Chen, P. Wang, L. Yue, Y. Zhang, and T. Jia, "Anomaly detection in surveillance video based on bidirectional prediction," *Image and Vision Computing*, vol. 98, Article ID 103915, 2020.
- [5] H. Xiang, J. Wang, K. Ramamohanarao, Z. Salcic, W. Dou, and X. Zhang, "Isolation forest based anomaly detection framework on non-IID data," *IEEE Intelligent Systems*, vol. 36, no. 3, pp. 31–40, 2021.
- [6] H. Mu, R. Sun, G. Yuan, and G. Shi, "Positive unlabeled learning-based anomaly detection in videos," *International Journal of Intelligent Systems*, vol. 36, no. 8, pp. 3767–3788, 2021.
- [7] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara, "Latent space autoregression for novelty detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 481–490, Los Angeles, CA, USA, June 2019.
- [8] Z. Cheng, S. Wang, P. Zhang, S. Wang, X. Liu, and E. Zhu, "Improved autoencoder for unsupervised anomaly detection," *International Journal of Intelligent Systems*, vol. 36, no. 12, pp. 7103–7125, 2021.
- [9] B. Zong, "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proceedings of the International Conference on Learning Representations*, Vancouver, Canada, April 2018.
- [10] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *Proceedings of the International Symposium on Neural Networks*, pp. 189–196, Springer, Athens, Greece, October 2017.
- [11] F. U. M. Ullah, A. Ullah, K. Muhammad, I. U. Haq, and S. W. Baik, "Violence detection using spatiotemporal features with 3D convolutional neural network," *Sensors*, vol. 19, no. 11, p. 2472, 2019.
- [12] W. Ullah, A. Ullah, I. U. Haq, K. Muhammad, M. Sajjad, and S. W. Baik, "CNN features with bi-directional LSTM for real-time anomaly detection in surveillance networks," *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16979–16995, 2021.
- [13] D. Zhang, J. Han, G. Cheng, and M. H. Yang, "Weakly supervised object localization and detection: a survey," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 44, no. 9, pp. 5866–5885, 2021.
- [14] T. Hussain, K. Muhammad, W. Ding, J. Lloret, S. W. Baik, and V. H. C. de Albuquerque, "A comprehensive survey of multi-view video summarization," *Pattern Recognition*, vol. 109, Article ID 107567, 2021.
- [15] Y. Zhang, X. Liang, D. Zhang, M. Tan, and E. P. Xing, "Unsupervised object-level video summarization with online motion auto-encoder," *Pattern Recognition Letters*, vol. 130, pp. 376–385, 2020.
- [16] G. Pang, C. Yan, C. Shen, A. V. D. Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12173–12182, Seattle, WA, USA, June 2020.
- [17] H. Zhao, J. Jia, and V. Koltun, "Exploring self-attention for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10076–10085, Seattle, WA, USA, June 2020.
- [18] G. Zhang, X. Zhang, M. Bilal, W. Dou, X. Xu, and J. J. Rodrigues, "Identifying fraud in medical insurance based on blockchain and deep learning," *Future Generation Computer Systems*, vol. 130, pp. 140–154, 2022.
- [19] A. Markovitz, G. Sharir, I. Friedman, L. Zelnik-Manor, and S. Avidan, "Graph embedded pose clustering for anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10539–10547, Seattle, WA, USA, June 2020.
- [20] H. Park, J. Noh, and B. Ham, "Learning memory-guided normality for anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14372–14381, Seattle, WA, USA, June 2020.
- [21] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli, "Adversarially learned one-class classifier for novelty detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3379–3388, Salt Lake City, UT, USA, June 2018.
- [22] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection—a new baseline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6536–6545, Salt Lake City, UT, USA, June 2018.
- [23] P. Burlina, N. Joshi, and I. Wang, "Where's Wally now? Deep generative and discriminative embeddings for novelty detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11507–11516, Long Beach, CA, USA, June 2019.
- [24] T.-N. Nguyen and J. Meunier, "Anomaly detection in video sequence with appearance-motion correspondence," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1273–1283, Seoul, Korea, October 2019.
- [25] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proceedings of the IEEE*

- Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, Salt Lake, UT, USA, June 2018.
- [26] Y. Zhu and S. Newsam, “Motion-aware feature for improved video anomaly detection,” 2019, <https://arxiv.org/abs/1907.10211>.
- [27] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1237–1246, Long Beach, CA, USA, July 2019.
- [28] D. Zhang, W. Zeng, J. Yao, and J. Han, “Weakly supervised object detection using proposal-and semantic-level relationships,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, 2020.
- [29] W. Ullah, A. Ullah, T. Hussain, Z. A. Khan, and S. W. Baik, “An efficient anomaly recognition framework using an attention residual LSTM in surveillance videos,” *Sensors*, vol. 21, no. 8, p. 2811, 2021.
- [30] B. Wan, Y. Fang, X. Xia, and J. Mei, “Weakly supervised video anomaly detection via center-guided discriminative learning,” in *Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, London, UK, July 2020.
- [31] J. Zhang, L. Qing, and J. Miao, “Temporal convolutional network with complementary inner bag loss for weakly supervised anomaly detection,” in *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP)*, pp. 4030–4034, Taipei, Taiwan, September 2019.
- [32] D. Xu, R. Song, X. Wu, N. Li, W. Feng, and H. Qian, “Video anomaly detection based on a hierarchical activity discovery within spatio-temporal contexts,” *Neurocomputing*, vol. 143, pp. 144–152, 2014.
- [33] W. Luo, W. Liu, and S. Gao, “A revisit of sparse coding based anomaly detection in stacked rnn framework,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 341–349, Venice, Italy, October 2017.
- [34] W. Ullah, A. Ullah, T. Hussain et al., “Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data,” *Future Generation Computer Systems*, vol. 129, pp. 286–297, 2022.
- [35] W. Ullah, T. Hussain, Z. A. Khan, U. Haroon, and S. W. Baik, “Intelligent dual stream CNN and echo state network for anomaly detection,” *Knowledge-Based Systems*, vol. 253, Article ID 109456, 2022.
- [36] W. Liu, W. Luo, Z. Li, P. Zhao, and S. Gao, “Margin learning embedded prediction for video anomaly detection with a few anomalies,” in *Proceedings of the International Joint Conference on Artificial In*, pp. 3023–3030, Vienna, Austria, July 2019.
- [37] W. Ullah, T. Hussain, F. U. M. Ullah, M. Y. Lee, and S. W. Baik, “TransCNN: hybrid CNN and transformer mechanism for surveillance anomaly detection,” *Engineering Applications of Artificial Intelligence*, vol. 123, Article ID 106173, 2023.
- [38] W. Ullah, T. Hussain, and S. W. Baik, “Vision transformer attention with multi-reservoir echo state network for anomaly recognition,” *Information Processing & Management*, vol. 60, no. 3, Article ID 103289, 2023.
- [39] M. Pourreza, M. Salehi, and M. Sabokrou, “Ano-graph: learning normal scene contextual graphs to detect video anomalies,” 2021, <https://arxiv.org/abs/2103.10502>.
- [40] V. D. Lyakhovskiy, “Chebyshev polynomials and the proper decomposition of functions,” *Theoretical and Mathematical Physics*, vol. 200, no. 2, pp. 1147–1157, 2019.
- [41] J. Carreira and A. Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, Honolulu, Hawaii, July 2017.
- [42] S. Paul, S. Roy, and A. K. Roy-Chowdhury, “W-talc: weakly-supervised temporal activity localization and classification,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 563–579, Munich, Germany, September 2018.
- [43] M. Rashid, H. Kjellstrom, and Y. J. Lee, “Action graphs: weakly-supervised action localization with graph convolution networks,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 615–624, Snowmass, CO, USA, March 2020.
- [44] L. Shi, Y. Zhang, J. Cheng, and H. Lu, “Two-stream adaptive graph convolutional networks for skeleton-based action recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12026–12035, Long Beach, CA, USA, June 2019.
- [45] J. Wu, S. H. Zhong, and Y. Liu, “Dynamic graph convolutional network for multi-video summarization,” *Pattern Recognition*, vol. 107, Article ID 107382, 2020.
- [46] B. Zhao, H. Li, X. Lu, and X. Li, “Reconstructive sequence-graph network for video summarization,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, 2021.
- [47] T.-T. Nguyen, G. T. T. Nguyen, T. Nguyen, and D. H. Le, “Graph convolutional networks for drug response prediction,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 19, no. 1, pp. 146–154, 2022.
- [48] Z. Wang, M. Zhou, and C. Arnold, “Toward heterogeneous information fusion: bipartite graph convolutional networks for in silico drug repurposing,” *Bioinformatics*, vol. 36, pp. i525–i533, 2020.
- [49] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” *Advances in Neural Information Processing Systems*, vol. 29, pp. 3844–3852, 2016.
- [50] A. Mazari and H. Sahbi, “MLGCN: multi-laplacian graph convolutional networks for human action recognition,” in *Proceedings of the British Machine Vision Conference*, Cardiff, UK, November 2019.
- [51] S. Li, M. Pilipczuk, and M. Sorge, “Cluster Editing parameterized above the size of a modification-disjoint $\$ P_3 \$$ packing is para-NP-hard,” 2019, <https://arxiv.org/abs/1910.08517>.
- [52] J. Liu and J. Han, “Spectral clustering,” in *Data Clustering*, Chapman and Hall/CRC, Boca Raton, FL, USA, 2018.
- [53] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, “Learning temporal regularity in video sequences,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 733–742, Honolulu, Hawaii, July 2016.
- [54] Y. Liu, C.-L. Li, and B. Póczos, “Classifier two sample test for video anomaly detections,” in *Proceedings of the British Machine Vision Conference*, p. 71, Newcastle, UK, August 2018.
- [55] F. Sohrab, J. Raitoharju, M. Gabbouj, and A. Iosifidis, “Subspace support vector data description,” in *Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 722–727, IEEE, Beijing, China, August 2018.
- [56] D. Gong, “Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1705–1714, Long Beach, CA, USA, June 2019.

- [57] J. Wang and A. Cherian, "Gods: generalized one-class discriminative subspaces for anomaly detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8201–8211, Long Beach, CA, USA, June 2019.
- [58] F. Dong, Y. Zhang, and X. Nie, "Dual discriminator generative adversarial network for video anomaly detection," *IEEE Access*, vol. 8, pp. 88170–88176, 2020.
- [59] K. Doshi and Y. Yilmaz, "Any-shot sequential anomaly detection in surveillance videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 934–935, Washington DC, USA, December 2020.
- [60] Y. Tang, L. Zhao, S. Zhang, C. Gong, G. Li, and J. Yang, "Integrating prediction and reconstruction for anomaly detection," *Pattern Recognition Letters*, vol. 129, pp. 123–130, 2020.
- [61] Y. Chang, Z. Tu, W. Xie, and J. Yuan, "Clustering driven deep autoencoder for video anomaly detection," in *Proceedings of the European Conference on Computer Vision*, pp. 329–345, Springer, Glasgow, UK, November 2020.
- [62] T. Li, Z. Wang, S. Liu, and W.-Y. Lin, "Deep unsupervised anomaly detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3636–3645, Waikoloa, HI, USA, January 2021.
- [63] K. Doshi and Y. Yilmaz, "Online anomaly detection in surveillance videos with asymptotic bound on false alarm rate," *Pattern Recognition*, vol. 114, Article ID 107865, 2021.
- [64] S. Chandrakala, K. Deepak, and V. Lkp, "Bag-of-Event-Models based embeddings for detecting anomalies in surveillance videos," *Expert Systems with Applications*, vol. 190, Article ID 116168, 2022.
- [65] Y. Liu, J. Liu, J. Lin, M. Zhao, and L. Song, "Appearance-motion united auto-encoder framework for video anomaly detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, 2022.
- [66] A. Osman Tur, N. Dall'Asen, C. Beyan, and E. Ricci, "Exploring diffusion models for unsupervised video anomaly detection," 2023, <https://arxiv.org/abs/2304.05841>.
- [67] Y. Wang, T. Liu, J. Zhou, and J. Guan, "Video anomaly detection based on spatio-temporal relationships among objects," *Neurocomputing*, vol. 532, pp. 141–151, 2023.
- [68] V.-T. Le and Y.-G. Kim, "Attention-based residual autoencoder for video anomaly detection," *Applied Intelligence*, vol. 53, no. 3, pp. 3240–3254, 2023.
- [69] H. Sapkota and Q. Yu, "Bayesian nonparametric submodular video partition for robust anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3212–3221, Washington DC, USA, June 2022.
- [70] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4489–4497, Washington, DC, USA, July 2015.