Scene-Aware Context Reasoning for Unsupervised Abnormal Event Detection in Videos

Che Sun
Beijing Institute of Technology
Beijing, China
sunche@bit.edu.cn

Yunde Jia
Beijing Institute of Technology
Beijing, China
jiayunde@bit.edu.cn

Yao Hu
Alibaba Youku Cognitive and Intelligent Lab
Beijing, China
yaoohu@alibaba-inc.com

Yuwei Wu*
Beijing Institute of Technology
Beijing, China
wuyuwei@bit.edu.cn

ABSTRACT

In this paper, we propose a scene-aware context reasoning method that exploits context information from visual features for unsupervised abnormal event detection in videos, which bridges the semantic gap between visual context and the meaning of abnormal events. In particular, we build a spatio-temporal context graph to model visual context information including appearances of objects, spatio-temporal relationships among objects and scene types. The context information is encoded into the nodes and edges of the graph, and their states are iteratively updated by using multiple RNNs with message passing for context reasoning. To infer the spatio-temporal context graph in various scenes, we develop a graph-based deep Gaussian mixture model for scene clustering in an unsupervised manner. We then compute frame-level anomaly scores based on the context information to discriminate abnormal events in various scenes. Evaluations on three challenging datasets, including the UCF-Crime, Avenue, and ShanghaiTech datasets, demonstrate the effectiveness of our method.

CCS CONCEPTS

• **Computing methodologies → Scene anomaly detection**; *Activity recognition and understanding*.

KEYWORDS

Abnormal Event Detection; Visual Context; Context Reasoning; Spatio-temporal Context Graph

ACM Reference Format:
Che Sun, Yunde Jia, Yao Hu, and Yuwei Wu. 2020. Scene-Aware Context Reasoning for Unsupervised Abnormal Event Detection in Videos. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413887

1 INTRODUCTION

Detecting abnormal events in videos is a challenging problem due to diverse events, lack of training data, and highly contextual definition of abnormal events [8, 15, 29]. Several existing methods are devoted to learning normal spatio-temporal patterns of appearance and motion, and then detecting abnormal events by distinguishing the events from the normal patterns [4, 37]. They usually extract visual features based on an image [3, 6] or an isolated object region [8] in the image to learn the normal spatio-temporal patterns.

Psychological evidence shows that humans can recognize objects and scenes comprehensively through exploiting visual context information [1, 34], and a variety of computer vision tasks benefit from context information [7, 30, 32, 33]. Therefore, mining abundant context information beyond the image-level and object-level features is also critical to discriminating abnormal events in a video, where visual context highly influences the way to determine abnormal events. Taking vehicle stopping as an example, the vehicle stopping at traffic junctions is considered as a normal event, but stopping on highways is an abnormal event. False detection will occur if ignoring the context information of traffic junctions and highways.

Prior to our work, all the efforts of context-based abnormal event detection manually pre-define the collections of context based on the human experience. For example, several methods develop context models of specific relationships among objects, such as support relationships [2], co-occurrence and geometric relationships [23], and so on. In addition to the relationship context, recent work [14] four specific types of scene context for anomaly prediction. These methods treat behaviors that deviate from the pre-defined collections of context as abnormal, meaning that the correctness and completeness of the collections are crucial to the performance of abnormal event detection. Unfortunately, it is impossible to manually pre-define the collections that take all possible context information involved in events into account, because in many cases the definition of context-related events is diverse, constantly changing, and unpredictable [24].

In this paper, we propose to automatically learn context information from data rather than manually pre-define contextual contents.

*Corresponding Author.

MM '20, October 12–16, 2020, Seattle, WA, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413887

To this end, we perform a context reasoning method to mine high-level context information from low-level visual features of data, which bridges the semantic gap between visual context and the meaning of abnormal events. Specifically, we construct a spatial context graph (SC Graph) from a single frame to learn appearances of objects and spatial relationships among different objects. The appearances and relationships are encoded into representations of the nodes and edges of the graph, respectively. Furthermore, the SC Graphs are input into the structural recurrent neural network (structural-RNN) for building a spatio-temporal context graph (STC Graph), where temporal dynamics of each object are encoded as representations of temporal edges of the STC graph. In order for context reasoning, we iteratively update the states of the nodes and edges of the STC graph using a mean-field like procedure to infer semantic context from visual features, and then discriminate abnormal events based on the reasoned context.

Since abnormal events are usually rare [6, 21], we introduce a scene clustering strategy to infer the spatio-temporal context graph in various scenes in an unsupervised manner. We develop a graph-based deep Gaussian mixture model to divide scenes into groups. Events in a group are regarded to be normal for the group and abnormal for other groups. The labeled normal and abnormal events are used to infer the spatio-temporal graph for each group. During detecting abnormal events, the scenes of events are identified with the scene clustering, and abnormal events are discriminated in the clustered scenes.

We conduct experiments on the UCF-Crime [31], Avenue [18] and ShanghaiTech [19] datasets to verify the effectiveness of our method. The UCF-Crime dataset is a large-scale dataset of real-world surveillance videos that contains diverse abnormal events in both indoor and outdoor scenes. Our method significantly improves the performance of the state-of-the-art unsupervised methods. Compared with supervised methods, we still gain comparable results in an unsupervised manner. On the Avenue and ShanghaiTech datasets, we have achieved relatively obvious improvement compared with the existing methods.

In summary, this paper makes the following contributions.

- We propose a novel method of abnormal event detection in videos via scene-aware context reasoning that mines context information from visual features of data to bridge the semantic gap between visual context and the meaning of abnormal events. To the best of our knowledge, it is the first work that uses the scene-aware context reasoning for the problem.
- We build a spatio-temporal context graph for context encoding and reasoning, which takes full advantage of context information for discriminating abnormal events.
- We develop a graph-based deep Gaussian mixture model for scene clustering to identify scene types. The scene clustering contributes to detecting context-related and unknown abnormal events in different scenes.

2 RELATED WORK

Recently, many researchers have focused on abnormal event detection in videos [10, 16, 28, 39]. Existing methods are broadly categorized as classification-based, reconstruction-based, and context-based.

Classification-based methods usually use one or more oneclass classifiers trained by normal events, and then detect abnormal events through classifier scores [35, 39]. For example, Ionescu *et al.* [35] extracted deep features using convolutional neural networks (CNN) and adopted a one-class Support Vector Machines (SVM) model to classify normal and abnormal events. Xu *et al.* [37] used multiple one-class SVM models based on fused deep appearance and motion features to predict anomaly scores. Ionescu *et al.* [8] formalized abnormal event detection as a one-versus-rest binary classification problem by learning spatio-temporal features and clustering the training samples into normality clusters. In contrast, we perform scene clustering to identify scene types, and further predict anomaly scores with multiple dummy binary classifiers in the clustered scenes.

Reconstruction-based methods learn normal patterns and distinguish abnormal events through reconstruction errors. Hasan *et al.* [6] used one fully connected auto-encoder and another end-to-end convolutional auto-encoder to reconstruct normal events, respectively, to learn the regular dynamics. Chong and Tay [3] proposed a spatio-temporal auto-encoder for abnormal event detection. Gong *et al.* [5] proposed a Memory-augmented Deep Auto-encoder (MemAE) and detect abnormal events with higher reconstruction errors. These methods focus on designing a good reconstruction model such as auto-encoders to model spatio-temporal normal patterns of the whole frame or per-object in videos. Different from these methods, we take full advantage of context cues by encoding them into a spatio-temporal context graph, and perform context reasoning and abnormal event detection through iteratively updating the states of the graph with multiple RNNs.

There are also several methods that use context models to improve the performance of abnormal event detection [2, 14, 25]. Choi *et al.* [2] modeled support relationships between objects to detect “out-of-context” objects and scenes. Zhu *et al.* [41] defined a set of six context attributes related to the scene and involved objects in normal events, and detect abnormal events that deviate from the defined set. Leach *et al.* [14] modeled context information based on social scenes of four types for abnormal event detection. These methods manually pre-define regulation sets of context based on the human observation, and may not be applicable to different unknown abnormal events in various scenes. Differently, our method performs context reasoning to automatically mine high-level context information from low-level visual features of data, which can be used to discriminate abnormal events in various scenes.

3 METHOD

We present a scene-aware context reasoning method for abnormal event detection by constructing a spatio-temporal context graph in various scenes. As depicted in Figure 1, we build a spatial context graph for each frame to model the appearance of objects and the spatial relationships of different objects. Then the spatial context graphs are input into the structural-RNN [9] to learn the temporal dynamics of each object for constructing the spatio-temporal context graph. Furthermore, we introduce a scene clustering strategy to identify scene types and infer the spatio-temporal context

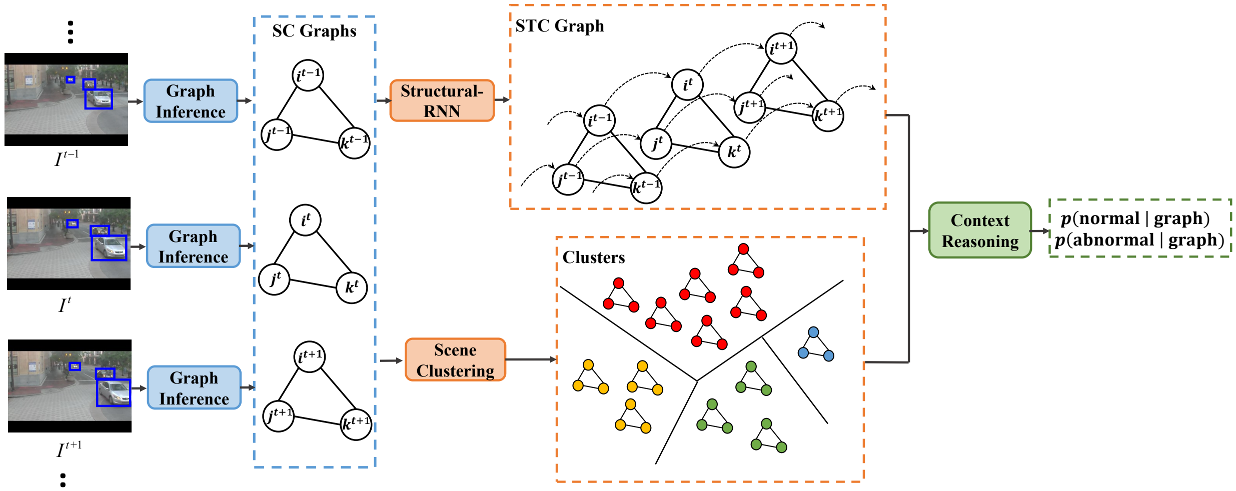

Figure 1: The framework of our method. The spatial context graph (SC Graph) is inferred from each frame I with object bounding boxes, where i, j, k denote the graph nodes and the superscripts $(t-1), t, (t+1)$ refer to timestamps of frames. The spatio-temporal context graph (STC Graph) is constructed by temporally modeling the SC Graph using the structural-RNN. Meanwhile, the scene clustering of the SC Graph is executed to identify different scene types. The reasoning model is made upon the STC graph to discriminate abnormal events in various scenes.

graph model in an unsupervised manner. Finally, context information is exploited for distinguishing abnormal events based on the spatio-temporal context graph and scene clustering.

3.1 Spatio-temporal Context Graph

The visual context is encoded into the spatio-temporal context graph (STC Graph), where the nodes are used to describe appearances of objects, the spatial edges denote the spatial relationships among objects, and the temporal edges represent the temporal dynamics of each object in videos. The STC Graph is built to infer the semantics of the individual object (nodes of the graph), spatio-temporal relationships (edges of the graph) and the scenes of events (the whole graph), which are used to detect point anomalies, contextual anomalies, and collective anomalies, respectively. We formulate abnormal event detection as the construction and inference of spatio-temporal context graph in various scenes, and perform context reasoning by iteratively updating the states of the nodes and edges of the graph representations.

3.1.1 Formulation. Given a video V with T frames $[I^1, I^2, \cdots, I^T]$, the region proposal network (RPN) [27] generates object bounding boxes for each frame, where top-K bounding boxes are selected in the t-th frame as B^t. The whole frame is also considered as an extra bounding box. For each frame, we build a spatial context graph (SC Graph) based on the image with the k bounding boxes. The node v_i of the SC Graph represents the i-th object, and the edge $e_{i,j}$ denotes the relationship between the i-th object and the j-th object. Each node and edge is assigned a "normal" or "abnormal" label that will be predicted by inferring the graph.

For the t-th frame, we define the label of the i-th object as y_i^t, and the label of the spatial relationship between the i-th and j-th objects ($i \neq j$) as $y_{i,j}^t$. All these labels are binary, i.e., 0 for normal objects (or relationships) and 1 for abnormal objects (or relationships). The set of all anomaly labels in the t-th frame is defined as $\mathbf{y}^t = \{y_i^t, y_{i,j}^t | i, j = 1, 2, \ldots, K; j \neq i\}$. The generation of the SC Graph is formulated as $\arg\max_{\mathbf{y}^t} P(\mathbf{y}^t | I^t, B^t)$, where

$$P(\mathbf{y}^t | I^t, B^t) = \prod_{i,j \in K} \prod_{j \neq i} P(y_i^t, y_{i,j}^t, | I^t, B^t). \tag{1}$$

Then we integrate temporal information of multiple SC Graphs into an STC Graph using the structural-RNN. A node v_i at the t-th frame is only connected to the same node v_i in the $(t+1)$-th frame with a temporal edge $e_{i,i}$, whose relationship label is defined as $y_{i,i}^t$. The final probability function is given by

$$P(\mathbf{y} | V, B) = \prod_{t \in T} \prod_{i,j \in K} \prod_{j \neq i} P(y_i^t, y_{i,j}^t, y_{i,i}^t | V, B), \tag{2}$$

where $\mathbf{y} = \{y_i^t, y_{i,j}^t | t = 1, 2, \ldots, T; i, j = 1, 2, \ldots, K\}$ denotes the set of all anomaly labels in a video.

3.1.2 Graph Inference. The semantics of context can be inferred through iteratively updating the states of the nodes and edges of the graph representations, where we adopt a graph inference method of the mean-field [26, 38]. The probability function $P(\mathbf{y}|\cdot)$ in Eq. (2) is approximated to $Q(\mathbf{y}|\cdot)$ that is decided by the current state of each node and each edge. Specifically, we use the states h_i^t and $h_{i,j}^t$ to represent the current states of the i-th node and the edge between the i, j-th nodes in the t-th frame, respectively. In Eq. (2), the probability distribution of the i-th node $P(y_i^t|\cdot)$ depends on states of all nodes and edges $h_i^t, h_{i,j}^t$, where $t = 1, 2, \cdots, T; i, j = 1, 2, \cdots, K$. The mean-field approximation probability distribution $Q(y_i^t|\cdot)$ only depends on its current states (i.e. $Q(y_i^t|\cdot) = Q(y_i^t|h_i^t)$). The probability distributions of edges are also approximated in a similar way.

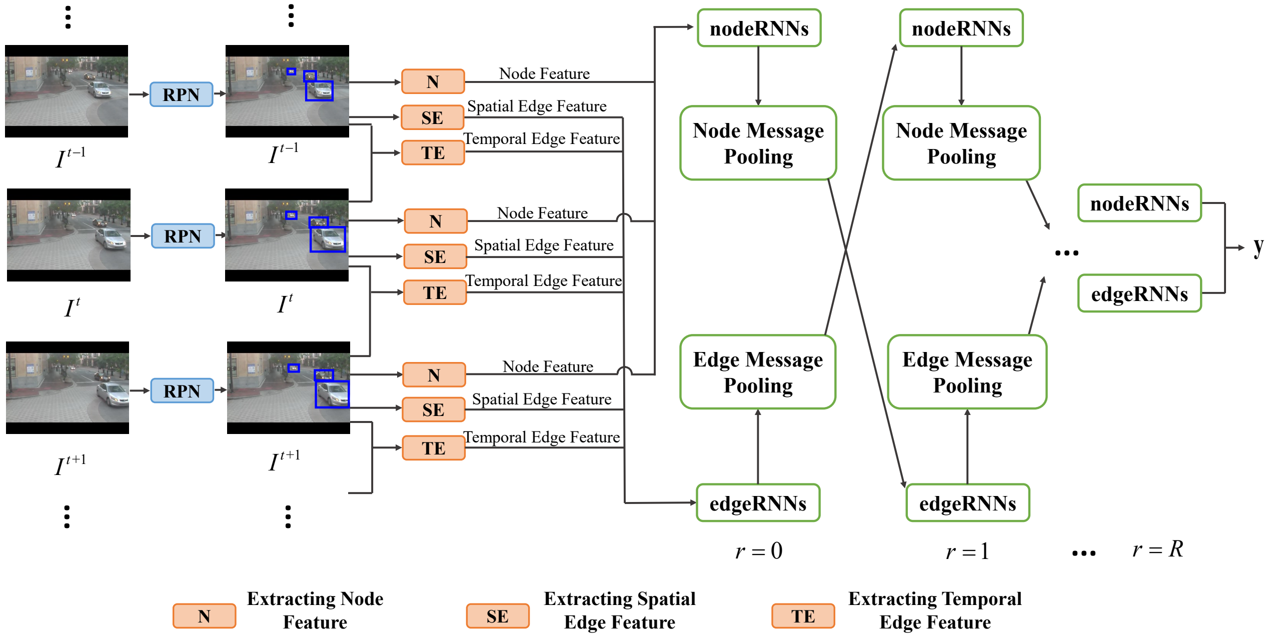

Figure 2: Illustration of the STC Graph inference. Object bounding boxes are obtained from raw frames via the region proposal network (RPN). Then the nodes and edges features are extracted by three feature extraction modules: the module *N* extracts node features, the module *SE* extracts spatial edge features, and the module *TE* extracts temporal edge features. These features are regarded as the initial input of the nodeRNNs and edgeRNNs for graph inference. The message passing method is introduced to update the hidden states of RNNs. When *R* times iteration of RNNs ($r = R$), the nodeRNNs and the edgeRNNs output the prediction of anomaly labels y.

With the mean-field distribution, the probability function In Eq. (2) is transferred into

$$\begin{aligned} Q(\mathbf{y}|V,B) = &\prod_{t=1}^{T}\prod_{i=1}^{K} Q(y_i^t|h_i^t)Q(h_i^t|f_i^t) \\ &\times \prod_{j=1,j\neq i}^{K} Q(y_{i,j}^t|h_{i,j}^t)Q(h_{i,j}^t|f_{i,j}^t) \\ &\times Q(y_{i,i}^t|h_{i,i}^t)Q(h_{i,i}^t|f_{i,i}^t), \end{aligned} \tag{3}$$

where f_i^t is the initial appearance feature of the i-th object in the t-th frame, $f_{i,j}^t$ is the initial spatial relationship feature between the i-th object and j-th object, and $f_{i,i}^t$ is the initial temporal relationship feature of the i-th object in the t-th and $(t+1)$-th frames. f_i^t is extracted from the bounding box of the object with the RPN, and $f_{i,j}^t$ is extracted from the union bounding box over its objects. We concatenate f_i^t and f_i^{t+1} as $[f_i^t, f_i^{t+1}]$ to integrate temporal information of the i-th object. Then a learnable matrix $\mathbf{W}_d$ is used to reduce the dimensionality of the concatenated feature by half to generate the temporal relationship feature $f_{i,i}^t = \mathbf{W}_d[f_i^t, f_i^{t+1}]$ in the t-th and $(t+1)$-th frames.

Inspired by [38], we compute the states in $Q(\mathbf{y}|\cdot)$ with multiple RNNs, and the hidden states of the RNNs are considered as the current states h_i^t, $h_{i,j}^t$, and $h_{i,i}^t$ used in $Q(\mathbf{y}|\cdot)$. As shown in Figure 2, the hidden states of nodes are modeled by nodeRNNs with one identical set of parameters, and the hidden states of spatio-temporal edges are modeled by edgeRNNs with another set of parameters. The spatial edges and temporal edges are modeled by the same edgeRNNs. The nodeRNNs and edgeRNNs iteratively update the states of the nodes and edges in the STC Graph to infer the semantics of context from visual features. During the iterative update, we also adopt Message Passing to improve the inference efficiency. The r-th iteration of nodeRNNs is formulated as

$$\begin{aligned} m_i^{t,r} &= \sum_j \sigma(\mathbf{W}_n^{1\top}[h_i^{t,r}, h_{i,j}^{t,r}])h_{i,j}^{t,r} \\ &\quad + \sum_j \sigma(\mathbf{W}_n^{2\top}[h_i^{t,r}, h_{j,i}^{t,r}])h_{j,i}^{t,r}, \\ h_i^{t,r+1} &= RNN_{node}(m_i^{t,r}, h_i^{t,r}), \end{aligned} \tag{4}$$

where σ denotes activation function (sigmoid), $[\cdot,\cdot]$ represents concatenating the vectors. $\mathbf{W}_n^1$ and $\mathbf{W}_n^2$ are learnable parameters. RNN_{node} denotes the recursive function. Similarly, the r-th iteration of edgeRNNs is formulated as

$$\begin{aligned} m_{i,j}^{t,r} &= \sigma(\mathbf{W}_e^{1\top}[h_{i,j}^{t,r}, h_i^{t,r}])h_i^{t,r} \\ &\quad + \sigma(\mathbf{W}_e^{2\top}[h_{i,j}^{t,r}, h_j^{t,r}])h_j^{t,r}, \\ h_{i,j}^{t,r+1} &= RNN_{edge}(m_{i,j}^{t,r}, h_{i,j}^{t,r}), \end{aligned} \tag{5}$$

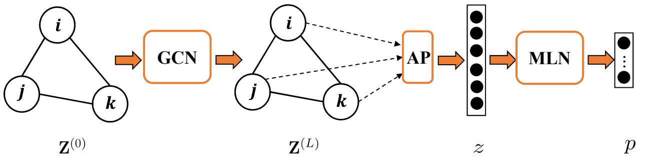

Figure 3: Illustration of the graph-based deep Gaussian mixture model. The graph representations $\mathbf{Z}^{(0)}$ are modeled by the graph convolutional network (GCN) and the average pooling layer (AP), generating the vector representations z of the graph. The vector representations are clustered to divide the scenes into groups using the GMM modeled by the multi-layer network (MLN).

where $\mathbf{W}_e^1$ and $\mathbf{W}_e^2$ are learnable parameters. RNN_{edge} denotes the recursive function.

3.2 Scene Clustering

It is common that a normal event in some scenes becomes abnormal in others, which means that identifying scene types is vital to understanding abnormal events. Hence we introduce a scene clustering strategy to identify scene types and meanwhile infer the STC Graph in an unsupervised manner.

Humans can easily identify the scene types from a single image, so we discriminate different scenes through clustering the SC Graphs in a static frame. The scene clustering divides the scenes of events into groups. All events in a group are regarded as normal events for the group and dummy anomalies for the other groups. These normal and "abnormal" events are used to infer the context graph for each group and detect abnormal events.

We build a graph-based deep Gaussian mixture model (GMM) for scene clustering. As shown in Figure 3, we first construct the SC Graph using the pre-trained model in [38] on Visual Genome dataset [13]. The outputs of the final layer of nodeRNNs are extracted as the node features $\mathbf{X} \in \mathbb{R}^{K \times D}$, where K is the number of nodes (bounding boxes) and D is the dimension of features. Then the graph convolutional network (GCN) [11] is utilized to model the graph by two graph convolutional layers

$$\mathbf{Z}^{(l)} = \sigma\left(\mathbf{A}\mathbf{Z}^{(l-1)}\mathbf{W}_c^{(l)}\right), \tag{6}$$

where $\mathbf{A} \in \mathbb{R}^{K \times K}$ is the adjacency matrix, $\sigma(\cdot)$ represents an activation function (relu in our paper), $\mathbf{Z}^{(l)}$ is the feature representation of nodes in the l-th layer, and $\mathbf{Z}^{(0)} = \mathbf{X}$. $\mathbf{W}_c^{(l)}$ is the layer-specific trainable weight matrix. The generated SC Graph is full-connected, so the elements of matrix A are all set to 1.

The GCN outputs another graph with node representations. We perform average pooling operation on the outputs of the GCN to get the vector representation z of the graph. A deep network of the GMM [42] is adopted for clustering on the normalized vector representations z. The GMM estimates the parameters of the mixture-component distribution ϕ, mixture means μ, and mixture covariance Σ to realize maximum likelihood function or minimum energy function. The deep network of the GMM is a multi-layer network (MLN) to output the mixture membership and calculate these parameters. We set the number of components to M, and set the number of batches to N. The network is

$$\begin{aligned} p &= MLN(z;\theta), \\ \hat{\gamma} &= softmax(p), \end{aligned} \tag{7}$$

where θ denotes the parameters of MLN and $\hat{\gamma} \in \mathbb{R}^{N \times M}$ is the soft mixture-component membership prediction. The estimated parameters of the m-th component of the GMM are

$$\begin{aligned} \hat{\phi}_m &= \sum_{i=1}^{N} \frac{\hat{\gamma}_{i,m}}{N}, \\ \hat{\mu}_m &= \frac{\sum_{i=1}^{N} z_i \hat{\gamma}_{i,m}}{\sum_{i=1}^{N} \hat{\gamma}_{i,m}}, \\ \hat{\Sigma}_m &= \frac{\sum_{i=1}^{N} \hat{\gamma}_{i,m}(z_i - \hat{\mu}_m)(z_i - \hat{\mu}_m)^T}{\sum_{i=1}^{N} \hat{\gamma}_{i,m}}, \end{aligned} \tag{8}$$

where $\hat{\phi}_m$, $\hat{\mu}_m$ and $\hat{\Sigma}_m$ are mixture-component distribution, mean, covariance for the m-th component, respectively. The sample energy function is used to estimate parameters as

$$E = -\log\left(\sum_{m=1}^{M} \hat{\phi}_m \frac{\exp\left(-\frac{1}{2}(z - \hat{\mu}_m)^T \hat{\Sigma}_m^{-1}(z - \hat{\mu}_m)\right)}{\sqrt{|2\pi\hat{\Sigma}_m|}}\right), \tag{9}$$

where $|\cdot|$ calculate the determinant of a matrix.

We fix the parameters of the feature extraction network, and optimize the parameters of the GCN and the deep network of the GMM simultaneously using the loss function of

$$\mathcal{L}_{\text{clu}} = \frac{1}{N}\sum_{i=1}^{N} E + \lambda_1 \sum_{m=1}^{M}\sum_{j=1}^{d} \frac{1}{\hat{\Sigma}_{m,j,j}}, \tag{10}$$

where λ_1 is the trade-off parameter and d is the dimension of z. The second term of Eq. (10) penalizes small values on the diagonal entries to avoid the singularity problem. In this paper, λ_1 is set to 0.005.

3.3 Model Training

Through the scene clustering, we have divided the context scenes into groups, and have labeled the objects and relationships in videos. The labeled data is used for the STC Graph inference, where the nodes predict whether the objects are abnormal, and the edges predict whether the relationships are abnormal. In our network, all learnable model parameters are optimized via the back propagation (BP) algorithm under an end-to-end condition. To this end, we use regularized cross-entropy loss function to maximize the probability in Eq. (3). For each group, we calculate the probability of being normal or abnormal for the node v_i and the edge $e_{i,j}$ in the t-th frame by

$$\begin{aligned} P(y_i^t|V,B) &= softmax(MLN(h_i^t)), \\ P(y_{i,j}^t|V,B) &= softmax(MLN(h_{i,j}^t)), \end{aligned} \tag{11}$$

where MLN represents a neural network with two fully-connected layers. For simplicity, we write the anomaly probability $P(y_i^t|V,B)$ and $P(y_{i,j}^t|V,B)$ as p_i^t and $p_{i,j}^t$, respectively. The loss function of graph inference is given by

$$\begin{aligned}\mathcal{L}^m =& \frac{1}{TK}\sum_{t=1}^{T}\sum_{i=1}^{K}\mathcal{L}_{\text{cls}}(y_i^t, p_i^t)\\ &+\frac{1}{TK^2}\sum_{t=1}^{T}\sum_{i=1}^{K}\sum_{j=1}^{K}\mathcal{L}_{\text{cls}}(y_i^t, p_{i,j}^t) + \lambda_2\|\mathbf{W}_m\|_1,\end{aligned} \tag{12}$$

where $\mathcal{L}_{\text{cls}}(\cdot,\cdot)$ is the binary cross entropy loss function, and $\mathbf{W}_m$ denotes the parameters of the *MLN*. λ_2 is the regularization term and is set to 0.0001 in this paper. The superscript m of $\mathcal{L}^m$ denotes that the classifier is trained on the m-th group. In the group, the labels of all objects y_i^t and relationships $y_{i,j}^t$ are set to normal, and we randomly sample data from other groups and label them as dummy anomalies to train the graph model.

3.4 Anomaly Score

For each group, an independent object classifier and an independent relationship classifier are trained to generate classification scores of objects and relationships. The final anomaly scores are calculated based on these classification scores to discriminate abnormal events in videos.

In the testing procedure, with one forward pass, the classification scores $P^m(y_i^t|V,B)$ and $P^m(y_{i,j}^t|V,B)$ of test data in the m-th group are calculated by Eq. (11). The lowest classification score in all scene groups is used as the anomaly score:

$$\begin{aligned} s_i^t &= \min\{P^m(y_i^t|V,B)|m=1,2,\cdots,M\},\\ s_{i,j}^t &= \min\{P^m(y_i^t|V,B)|m=1,2,\cdots,M\},\end{aligned} \tag{13}$$

where s_i^t and $s_{i,j}^t$ are the anomaly scores of objects and relationships, respectively, and M is the number of groups. These anomaly scores correspond to detecting individual anomalies and group anomalies, respectively. To obtain frame-level detection, we regard the highest score of all objects and relationships in a frame as the frame-level anomaly score. Since the anomaly score should vary smoothly among frames, we enforce temporal smoothness of the final frame-level anomaly scores using a Gaussian filter.

4 EXPERIMENTS

4.1 Datasets

We evaluate our method on the UCF-Crime [31], Avenue [18], and ShanghaiTech [19] datasets.

The **UCF-Crime** is a large-scale dataset of real-world surveillance videos with 13 types of abnormal events in diverse scenes, which consists of 1610 training videos and 290 testing videos. We select all normal training videos to train our network.

The **Avenue** dataset has 16 training and 21 testing videos with 35240 frames, totally. Each video lasts about 2 minutes long. Abnormal events are running, walking in opposite direction, throwing objects, and loitering.

The **ShanghaiTech** dataset contains 13 scenes with complex light conditions and various viewpoints. This dataset has 130 abnormal events and over 270, 000 training frames.

4.2 Implementation Details

We follow the experimental setting in [38] to construct the SC Graph. We use the MS COCO-pretrained RPN [27] that adopts the VGG-16 network as the backbone to generate object bounding boxes and extract visual features. The 512-dimensional feature vectors of bounding boxes generated by the RPN are used as the initial visual features of appearances f_i^t and relationships $f_{i,j}^t$. The hidden states $h_i^t, h_{i,j}^t$ of RNNs are also 512-dimensional vectors. Considering the density of objects in videos and computational efficiency, we directly use the RPN detector and the KLT tracker [22] to generate the top-K (10 in this paper) ground-truth bounding boxes of objects, and ignore the regression of the bounding box offsets. The KLT tracker is used to connect each object in adjacent frames for modeling temporal relationships of the STC graph. During training, a sliding window of 10 frames is sampled to construct the STC Graph, and the batch size is set to 16. We choose the RMSprop optimizer with a 0.001 learning rate to train the graph model.

When performing scene clustering, we randomly select one frame from each sliding window to generate the SC Graph. The clustering result of each frame is used to label the corresponding sliding window. The graph-based deep Gaussian mixture model is built by three graph convolutional layers and two fully-connected layers. In particular, the graph convolutional network runs with GC(512, 128, ReLU)-Drop(0.5)-GC(128, 32, ReLU)-Drop(0.5)-GC(32, 4, none). The architecture of the fully-connected network is FC(4, 32, ReLU)-Drop(0.5)-FC(32, 10, softmax). Layer(a, b, f) means a graph convolutional (GC) layer or fully-connected (FC) layer, where the size of the layer-specific trainable weight matrix is $a \times b$ and the activation function is f. Drop(p) refers to a dropout layer with the parameter p. During clustering, the number of clustering centers M denotes the preset number of scene types, and we set M to 10 to cover common scenes (e.g., campuses, highways, subways, etc). The training batch of scene clustering is set to 1024, and we use the RMSprop optimizer with a 0.0001 learning rate to train the clustering model.

4.3 Evaluation Metric

We evaluate our method based on the frame level, and compute anomaly scores of frames. The ROC curve is applied by gradually changing the threshold of anomaly scores. The corresponding Area Under Curve (AUC $\Uparrow$) and Equal Error Rate (EER $\Downarrow$) are also used for evaluation. In addition, the false alarm rate is introduced to evaluate the probability of misclassification. A higher AUC value, a lower EER value or a lower false alarm rate value indicate a better performance of abnormal event detection.

4.4 Comparisons

4.4.1 Results on the UCF-Crime Dataset. As Table 1 and Table 2 depicted, we report the performance of the AUC and the false alarm rate of our method compared with several existing unsupervised and supervised methods on the UCF-Crime dataset. We re-implement the work of Ionescu *et al.* [8], and replace the detector they used with the RPN detector for a fair comparison. The performances of other compared methods are taken from [31]. Our method significantly improves the performance of the state-of-the-art unsupervised method, gaining an improvement of 7.2% and

Table 1: Abnormal event detection results compared with unsupervised methods on the UCF-Crime dataset. ↑ represents that higher scores are better, and ↓ indicates that lower is better.

Training	Method	AUC↑	False Alarm↓
Unsupervised	Lu *et al.* [18]	65.5%	3.1%
	Hasan *et al.* [6]	50.6%	27.2%
	Ionescu *et al.* [8]	61.6%	8.5%
	Ours	**72.7%**	**2.2%**

Table 2: Abnormal event detection results compared with supervised methods on the UCF-Crime dataset.

Training	Method	AUC↑	False Alarm↓
Supervised	SVM Baseline	50%	-
	Sultani *et al.* [31]	**75.41%**	**1.9%**
Unsupervised	Ours	72.7%	2.2%

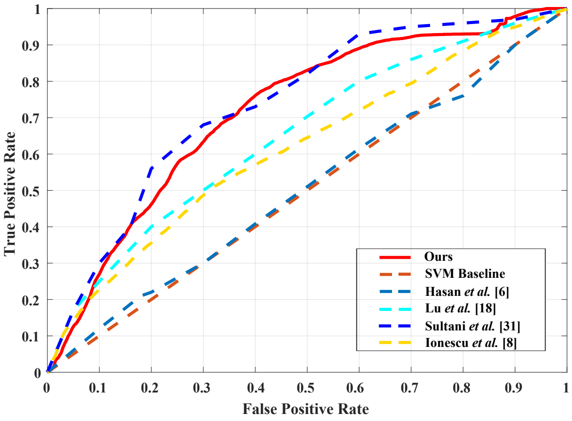

Figure 4: The ROC curves compared with several unsupervised and supervised methods on the UCF-Crime dataset.

0.9% on the AUC evaluation and the false alarm evaluation, respectively. This verifies the superiority of our method of detecting context-related abnormal events in various scenes. Our method is also comparable to the state-of-the-art supervised method [31]. Without video-level annotations, we achieve the comparable results of the AUC score and the false alarm rate, which demonstrates that our method can effectively detect unknown abnormal events in real-world applications.

As shown in Figure 4, we plot the ROC curves to evaluate our method. Our curve almost completely encloses the curves of the unsupervised methods, which means that our method outperforms the works of [6, 8, 18] at various thresholds. From the ROC curves, we observe that the performance of our method is comparable with the state-of-the-art supervised method [31]. Especially when the middle threshold is selected, our true positive rate slightly outperforms the work of [31], which demonstrates the effectiveness of our method.

Table 3: Abnormal event detection results in terms of frame-level AUC and EER on the Avenue dataset.

Method	AUC↑	EER↓
Hasan *et. al* [6]	70.2%	25.1%
Ionescu *et. al* [35]	80.6%	-
Chong and Tay [3]	80.3%	20.7%
Luo *et. al* [19]	81.7%	-
Liu *et. al* [17]	84.9%	-
Wang *et. al* [36]	85.3%	23.9%
Morais *et al.* [21]	86.3%	-
Ye *et al.* [40]	86.2%	-
Ours	**89.6%**	**21.1%**

Table 4: Abnormal event detection results in terms of frame-level AUC on the ShanghaiTech dataset.

Method	AUC↑	EER↓
Hasan *et. al* [6]	60.85%	-
Luo *et. al* [19]	68.00%	-
Liu *et al.* [17]	72.8%	-
Morais *et al.* [21]	73.4%	-
Ye *et al.* [40]	73.6%	-
Ours	**74.7%**	**28.6%**

4.4.2 *Results on the Avenue Dataset.* Table 3 shows that our method outperforms all existing methods on both the AUC and EER evaluations in the Avenue dataset. The state-of-the-art work of Morais *et al.* [21] achieves the AUC values of 86.4%, and our method gains a relatively significant improvement of 3.3%, which verifies that our method is effective and robust.

4.4.3 *Results on the ShanghaiTech Dataset.* We report experiment results on the ShanghaiTech dataset in Table 4. The ShanghaiTech dataset has complex scenes and various actions, which is recognized as a challenging benchmark of abnormal event detection. Our method outperforms the state-of-the-art methods on the ShanghaiTech dataset, which demonstrates the effectiveness of our method.

4.5 Ablation Study

In Table 5, we compare the contributions of different components in our method. "w/o spatial relationships" denotes removing the modeling of spatial relationships. Specifically, the STC Graphs are transformed to multiple object-centric chains across multiple frames, which are modeled by RNNs. "w/o temporal relationships" represents that we carry out the reasoning on the SC Graph. "w/o relationships" means that we use two fully-connected layers to model each object in isolation. In the above three cases, we perform the same scene clustering. "w/o scene clustering" refers to removing the scene clustering and only using a one-class classifier to classify all normal events for detecting abnormal events. From Table 5, it is interesting to observe that: (1) when discarding the spatial relationships, temporal relationships or spatio-temporal relationships, the performance of the AUC is reduced by 5.1%-14.7%, which verifies

Table 5: The AUC and false alarm results of different components of our method on the UCF-Crime dataset.

Method	AUC↑	False Alarm↓
w/o spatial relationships	61.8%	4.7%
w/o temporal relationships	67.6%	3.1%
w/o relationships	58.0%	13.4%
w/o scene clustering	63.6%	6.5%
Ours	**72.7%**	**2.2%**

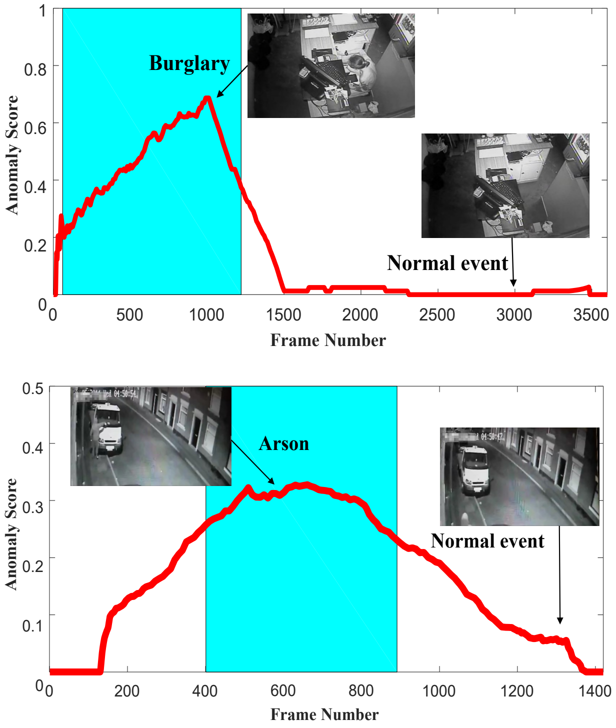

Figure 5: Qualitative results on the UCF-Crime dataset. Colored window shows the ground-truth labes of abnormal events.

the importance of the context relationships for discriminating abnormal events. (2) The performance is significantly improved by employing the scene clustering, demonstrating the effectiveness of discriminating different scenes for detecting abnormal events. The improvement also proves that the scene clustering is an effective unsupervised training strategy.

4.6 Event Count

To localize abnormal events according to anomaly scores, we select the local maximum in the time series of anomaly scores in a video. Specifically, we use the persistence1D algorithm [12] to identify the meaningful local maximum and span the region with a fixed temporal window. We follow the work of [6] to group nearby expanded local maximum regions if they overlap to obtain the final abnormal temporal regions, where abnormal events are localized into the temporal regions.

Table 6: The number of detected abnormal events and false alarm on the Avenue datasets. GT stands for groudtruth values of event count.

Method	True Positives↑	False Alarm↓
Medel *et. al* [20]	40	**2**
Hasan *et. al* [6]	45	4
Chong and Tay [3]	44	12
Ours	**46**	4

Table 6 shows the number of detected abnormal events and a false alarm on the Avenue dataset. Our method can detect abnormal events more precisely than the work of [6] and [3]. The False Alarm of our method is higher than that of [20], mainly because they select a small threshold of anomaly scores to detect abnormal events. Their method detects 40 true abnormal events, which is less than the 45 abnormal events detected by our method. The result demonstrates that our method can determine the temporal region of abnormal events more accurately, which makes it more practical in real scenes.

4.7 Qualitative Results

Figure 5 shows two examples of detected abnormal events, where colored windows show ground-truth labels of abnormal events, and curves represent anomaly scores of a portion of frames in the test video. The detected abnormal events are "burglary" and "arson" in the UCF-Crime dataset. From Figure 5, we can clearly see that the anomaly scores produced by our method match well with the ground-truth labels. Furthermore, there is a large score gap between normal and abnormal events, which validates the effectiveness of our method.

5 CONCLUSION

In this paper, we have presented a scene-aware context reasoning method for unsupervised abnormal event detection in videos. Context reasoning can explicitly mine high-level context information from low-level visual features. Through constructing the spatio-temporal context graph, the proposed method can explicitly model visual context by encoding appearances of objects and spatio-temporal relationships among objects into graph representations. In addition, we have developed a graph-based deep Gaussian mixture model for scene clustering that can effectively identify scene types and infer the spatio-temporal context graph in an unsupervised manner. Experiments on the UCF-Crime, Avenue and ShanghaiTech datasets demonstrate that our method outperforms existing state-of-the-art unsupervised methods, and is comparable to state-of-the-art supervised methods. In future work, we will exploit more fine-grained context information to extend our method from frame-level to pixel-level anomaly detection.

ACKNOWLEDGMENTS

This work was supported in part by the Natural Science Foundation of China (NSFC) under Grants No. 61702037 and No. 61773062, and Alibaba Group through Alibaba Innovative Research Program.

REFERENCES

[1] Bar and Moshe. 2004. Visual objects in context. *Nature Rev. Neurosci.* 5, 8 (2004), 617–629.

[2] Myung Jin Choi, Antonio Torralba, and Alan S Willsky. 2012. Context models and out-of-context objects. *Pattern Recognit. Lett.* 33 (2012), 853–862.

[3] Yong Shean Chong and Yong Haur Tay. 2017. Abnormal Event Detection in Videos Using Spatiotemporal Autoencoder. In *Proc. Adv. Neural Net.* 189–196.

[4] Yachuang Feng, Yuan Yuan, and Xiaoqiang Lu. 2016. Deep Representation for Abnormal Event Detection in Crowded Scenes. In *Proc. ACM Conf. Multimedia.* 591–595.

[5] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. 2019. Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection. In *Proc. IEEE Int. Conf. Comput. Vis.* 1705–1714.

[6] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. 2016. Learning Temporal Regularity in Video Sequences. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 733–742.

[7] Mahmudul Hasan, Sujoy Paul, Anastasios I. Mourikis, and Amit K. Roy-Chowdhury. 2020. Context-Aware Query Selection for Active Learning in Event Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 42, 3 (2020), 554–567.

[8] Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. 2019. Object-Centric Auto-Encoders and Dummy Anomalies for Abnormal Event Detection in Video. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 7842–7851.

[9] Ashesh Jain, Amir Roshan Zamir, Silvio Savarese, and Ashutosh Saxena. 2016. Structural-RNN: Deep Learning on Spatio-Temporal Graphs. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 5308–5317.

[10] Eric Jardim, Lucas A. Thomaz, Eduardo A. B. da Silva, and Sergio L. Netto. 2020. Domain-Transformable Sparse Representation for Anomaly Detection in Moving-Camera Videos. *IEEE Trans. Image Processing* 29 (2020), 1329–1343.

[11] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[12] Yeara Kozlov and Tino Weinkauf. 2015. Persistence1D: Extracting and filtering minima and maxima of 1d functions. http://www.csc.kth.se/~weinkauf/notes/persistence1d.html. (2015).

[13] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.* (2017), 32–73.

[14] Michael J. V. Leach, Ed P. Sparks, and Neil Martin Robertson. 2014. Contextual anomaly detection in crowded surveillance scenes. *Pattern Recognit. Lett.* 44 (2014), 71–79.

[15] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. 2013. Anomaly detection and localization in crowded scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* 36 (2013), 18–32.

[16] Kun Liu and Huadong Ma. 2019. Exploring Background-bias for Anomaly Detection in Surveillance Videos. In *Proc. ACM Conf. Multimedia.* 1490–1499.

[17] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. 2018. Future Frame Prediction for Anomaly Detection–A New Baseline. (2018), 6536–6545.

[18] Cewu Lu, Jianping Shi, and Jiaya Jia. 2013. Abnormal Event Detection at 150 FPS in MATLAB. In *Proc. IEEE Int. Conf. Comput. Vis.* 2720–2727.

[19] Weixin Luo, Wen Liu, and Shenghua Gao. 2017. A Revisit of Sparse Coding Based Anomaly Detection in Stacked RNN Framework. In *Proc. IEEE Int. Conf. Comput. Vis.* 341–349.

[20] Jefferson Ryan Medel and Andreas E. Savakis. 2016. Anomaly Detection in Video Using Predictive Convolutional Long Short-Term Memory Networks. *arXiv preprint arXiv:1612.00390* (2016).

[21] Romero Morais, Vuong Le, Truyen Tran, Budhaditya Saha, Moussa Reda Mansour, and Svetha Venkatesh. 2019. Learning Regularity in Skeleton Trajectories for Anomaly Detection in Videos. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 11996–12004.

[22] James Munkres. 1957. Algorithms for the assignment and transportation problems. *J. soc. ind. appl. Math.* 5 (1957), 32–38.

[23] Jongsuk Oh, Hong-In Kim, and Rae-Hong Park. 2017. Context-based abnormal object detection using the fully-connected conditional random fields. *Pattern Recognit. Lett.* 98 (2017), 16–25.

[24] Guansong Pang, Cheng Yan, Chunhua Shen, Anton van den Hengel, and Xiao Bai. 2020. Self-trained Deep Ordinal Regression for End-to-End Video Anomaly Detection. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 12173–12182.

[25] Sangdon Park, Wonsik Kim, and Kyoung Mu Lee. 2012. Abnormal object detection by canonical scene-based contextual model. In *Proc. Eur. Conf. Comput. Vis.* 651–664.

[26] Mengshi Qi, Yunhong Wang, Jie Qin, Annan Li, Jiebo Luo, and Luc Van Gool. 2020. stagNet: An Attentive Semantic RNN for Group Activity and Individual Action Recognition. *IEEE Trans. Circuits Syst. Video Techn.* 30, 2 (2020), 549–565.

[27] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 6 (2017), 1137–1149.

[28] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, and Reinhard Klette. 2017. Deep-Cascade: Cascading 3D Deep Neural Networks for Fast Anomaly Detection and Localization in Crowded Scenes. *IEEE Trans. Image Processing* 26, 4 (2017), 1992–2004.

[29] Hao Song, Che Sun, Xinxiao Wu, Mei Chen, and Yunde Jia. 2020. Learning Normal Patterns via Adversarial Attention-Based Autoencoder for Abnormal Event Detection in Videos. *IEEE Trans. Multimedia* 22, 8 (2020), 2138–2148.

[30] Wenfeng Song, Shuai Li, Tao Chang, Aimin Hao, Qinping Zhao, and Hong Qin. 2020. Context-Interactive CNN for Person Re-Identification. *IEEE Trans. Image Processing* 29 (2020), 2860–2874.

[31] Waqas Sultani, Chen Chen, and Mubarak Shah. 2018. Real-World Anomaly Detection in Surveillance Videos. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 6479–6488.

[32] Che Sun, Hao Song, Xinxiao Wu, and Yunde Jia. 2019. Learning Weighted Video Segments for Temporal Action Localization. In *Proc. Pattern Recognit. Comput. Vis.* 181–192.

[33] Jiangxin Sun, Jiafeng Xie, Jianfang Hu, Zihang Lin, Jianhuang Lai, Wenjun Zeng, and Wei-Shi Zheng. 2019. Predicting Future Instance Segmentation with Contextual Pyramid ConvLSTMs. In *Proc. ACM Conf. Multimedia.* 2043–2051.

[34] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to compose dynamic tree structures for visual contexts. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 6619–6628.

[35] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. 2017. Unmasking the Abnormal Events in Video. In *Proc. IEEE Int. Conf. Vis. Pattern Recognit.* 2895–2903.

[36] Siqi Wang, Yijie Zeng, Qiang Liu, Chengzhang Zhu, En Zhu, and Jianping Yin. 2018. Detecting Abnormality without Knowing Normality: A Two-stage Approach for Unsupervised Video Abnormal Event Detection. In *Proc. ACM Conf. Multimedia.* 636–644.

[37] Dan Xu, Yan Yan, Elisa Ricci, and Nicu Sebe. 2017. Detecting anomalous events in videos by learning deep representations of appearance and motion. *Comput. Vis. Image Underst.* 156 (2017), 117–127.

[38] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.* 5410–5419.

[39] Ke Xu, Tanfeng Sun, and Xinghao Jiang. 2020. Video Anomaly Detection and Localization Based on an Adaptive Intra-Frame Classification Network. *IEEE Trans. Multimedia* 22, 2 (2020), 394–406.

[40] Muchao Ye, Xiaojiang Peng, Weihao Gan, Wei Wu, and Yu Qiao. 2019. AnoPCN: Video Anomaly Detection via Deep Predictive Coding Network. In *Proc. ACM Conf. Multimedia.* 1805–1813.

[41] Yingying Zhu, Nandita M. Nayak, and Amit K. Roy-Chowdhury. 2013. Context-Aware Activity Recognition and Anomaly Detection in Video. *J. Sel. Topics Signal Processing* 7 (2013), 91–101.

[42] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Dae-ki Cho, and Haifeng Chen. 2018. Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. In *Proc. Int. Conf. Learn. Repren.*