Spatio-Temporal Feature Encoding for Traffic Accident Detection in VANET Environment

Zhili Zhou, *Member, IEEE*, Xiaohua Dong, Zhetao Li *Member, IEEE*, Keping Yu, *Member, IEEE*, Chun Ding, and Yimin Yang, *Senior Member, IEEE*

***Abstract*— In the Vehicular Ad hoc Networks (VANET) environment, recognizing traffic accident events in the driving videos captured by vehicle-mounted cameras is an essential task. Generally, traffic accidents have a short duration in driving videos, and the backgrounds of driving videos are dynamic and complex. These make traffic accident detection quite challenging. To effectively and efficiently detect accidents from the driving videos, we propose an accident detection approach based on spatio–temporal feature encoding with a multilayer neural network. Specifically, the multilayer neural network is used to encode the temporal features of video for clustering the video frames. From the obtained frame clusters, we detect the border frames as the potential accident frames. Then, we capture and encode the spatial relationships of the objects detected from these potential accident frames to confirm whether these frames are accident frames. The extensive experiments demonstrate that the proposed approach achieves promising detection accuracy and efficiency for traffic accident detection, and meets the real-time detection requirement in the VANET environment.**

***Index Terms*— Neural network, security communication, traffic accident detection, traffic safety, VANETs.**

I. Introduction

NOWADAYS, intelligent transportation systems (ITSs) have played an important role in enhancing road safety, traffic sustainability and efficiency, and vehicle management capacity. ITSs have become an indispensable part of our daily lives. As an essential paradigm in ITSs, Vehicular Ad hoc Networks (VANETs) offer connection and communication services among close vehicles and infrastructures on road by short-distance communication technologies [1], [2]. By monitoring the vehicle running statuses and surrounding conditions and sharing the information among those close vehicles, VANETs have been mainly used in the field of road safety in recent years. They mainly aim to reduce the possibility of traffic accidents and the losses caused by traffic accidents. There are three kinds of applications of VANETs: collision avoidance [3], [4], traffic sign notification [5], and incident management [6].

In the VANET environment, recognizing traffic accidents/abnormalities including the vehicle and pedestrian abnormalities in the natural driving videos captured by vehicle-mounted cameras is an essential task [7], [8]. Traffic accident/abnormality detection can be applied in a variety of applications such as road safety warning [9], [10], autonomous driving [11], traffic flow management [12], and pedestrian protection [13], [14]. In the past decades, many researchers have dedicated themselves to the study of accident/abnormality detection. The main challenges of accident/abnormality detection are the problems of long-tailed distribution and heterogeneous anomaly classes. To address these problems, many approaches have been proposed, which can be roughly classified into three categories: video-level, segment-level, and frame-level detection approaches.

The conventional accident/abnormality detection approaches are generally based on video-level [15], [16]. Generally, an accident/abnormality detector is directly learned from the whole traffic video using a variety of deep learning models for accident/abnormality detection. The deep learning models can be divided into two categories, *i.e.*, classifier-based and reconstruction error-based models. Classifier-based models assume that all normal instances are from the same category, while the basic assumption of reconstruction error-based models is that the reconstruction error for abnormal samples would be higher than that for normal samples. However, it is worth noting that the accidents/abnormalities usually have a short duration and follow a long-tailed distribution in driving videos, and the backgrounds of driving videos are dynamic and complex. Thus, it is very hard for these video-level approaches to achieve desirable performance for accident/abnormality detection.

To address the problem of long-tailed distribution in traffic accident/abnormality detection, some segment-level approaches [17], [18] have been proposed based on a weakly-supervised framework with Multiple Instances Learning (MIL). They use the high-level class labels to learn the anomaly score for each video segment directly. Some

Manuscript received August 7, 2021; revised December 6, 2021 and January 23, 2022; accepted January 26, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61972205, Grant 62032020, and Grant 62122032; in part by the Hunan Science and Technology Planning Project under Grant 2019RS3019; in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) Fund; and in part by the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET) Fund. The Associate Editor for this article was B. Gupta. *(Corresponding author: Zhili Zhou.)*

Zhili Zhou, Xiaohua Dong, and Chun Ding are with the Engineering Research Center of Digital Forecasts, Ministry of Education, School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: zhou_zhili@163.com; dongxiaohua@nuist.edu.cn; dingchun@nuist.edu.cn).

Zhetao Li is with the College of Information Science and Technology, Jinan University, Guangzhou 510632, China (e-mail: liztchina@hotmail.com).

Keping Yu is with the Global Information and Telecommunication Institute, Waseda University, Tokyo 169-8555, Japan (e-mail: keping.yu@aoni.waseda.jp).

Yimin Yang is with the Computer Science Department, Lakehead University, Thunder Bay, ON P7B 5E1, Canada (e-mail: yyang48@lakeheadu.ca).

Digital Object Identifier 10.1109/TITS.2022.3147826

other frame-level approaches have been proposed [19], [20], which use prediction models for vehicles and pedestrians' abnormality detection. These frame-level approaches usually construct the current frames by the deep learning models according to the previous frames. Then, they compute the construction error between the generated current frame and the original one to determine whether the current frame is an accident/abnormality frame. The frame-level approaches can more accurately localize the abnormalities in both the temporal and the spatial domains. However, these approaches generally need to construct each current frame and compute the corresponding construction error, which make them very time-consuming.

In summary, although these segment-level and frame-level approaches generally achieve superior performance than the video-level approaches to deal with the long-tailed distribution problem, they suffer from the issue of inefficiency in traffic accident/abnormality detection. That makes them inapplicable for traffic accident/abnormality detection in the VANET environment, in which real-time detection is usually required.

In this paper, we mainly focus on the detection of vehicle accidents/abnormalities in the VANET environment. To effectively and efficiently detect accident frames from the driving videos, we propose a novel frame-level accident detection approach based on spatio–temporal feature encoding with a multilayer neural network [21]. The framework of the proposed approach is based on the concept of coarse-to-fine detection, as shown in Fig. 1. Specifically, we first encode the two temporal features: the Histogram of Optical Flow (HOF) [22] features and the temporal ordinal features of frames as a temporal coding matrix by the multilayer neural network for clustering the frames. From the obtained frame clusters, we detect the border frames as the potential accident frames. Then, we encode the Convolutional Neural Network (CNN) [23] features and spatial relationships of the objects detected from these potential accident frames to confirm whether these frames are accident frames. The main contributions of this paper are summarized as follows:

1) A coarse-to-fine detection framework for traffic accident detection is proposed based on Spatial-temporal feature encoding. We first efficiently detect the potential accident frames, and then further confirm whether these frames are accident ones. That can ensure both high efficiency and accuracy in traffic accident detection. Also, the proposed approach can meet the real-time detection requirement in the VANET environment. Compared with the existing accident/abnormality detection approaches, our approach can effectively address a major challenge in the field of traffic accident detection—the long-tail distribution problem.

2) A novel unsupervised clustering approach is designed to detect abnormal frames, which can significantly reduce the time cost of temporal localization in videos. Thus, it can greatly improve time efficiency.

3) The superiority of the proposed approach has been demonstrated by extensive experiments. The experiment results show that the proposed approach increases the accuracy of temporal localization by 15.2% compared to the recent approaches. In our framework, we combine the temporal features with the spatial features to determine abnormal traffic video frames and identify the accident classes. Overall, the proposed approach increases the accuracy of accident detection significantly.

The remainder of this paper is organized as follows. Section II introduces the related work. Section III elaborates the proposed traffic accident detection approach. Section IV presents and discusses the experimental results. Conclusions are drawn in Section V.

II. Related Work

Traffic accident/abnormality detection including vehicle and pedestrian abnormality detection has received considerable attention in the field of intelligent transportation. In the past two decades, a lot of traffic accident/abnormality detection approaches have been proposed. These approaches can be roughly classified into three categories: video-level [15], [16], segment-level [17], [18], and frame-level detection approaches [19], [20]. Generally, these approaches focus on accident/abnormality detection in surveillance videos.

A. Video-Level Detection Approaches

In the early work, some video-level approaches have been proposed [15], [16], in which each video is treated as a whole for accident/abnormality detection. Specifically, each video is fed into a well-designed normality detection model. Such a model is learned by many training videos in a variety of normal instances. The videos are determined as abnormal ones if they deviate from the normality model. In general, these approaches treat the task of accident/abnormality detection as an outlier detection problem. The one-class classification models such as Support Vector Machines (SVM) [24] and Support Vector Data Description (SVDD) [16], [25] are popular for video-level accident/abnormality detection. However, these approaches require preprocessing, *i.e.*, feature extraction.

Some end-to-end approaches have been also proposed to directly learn a reconstruction model to distinguish abnormal instances. Specifically, these approaches first train a reconstruction model to reconstruct the feature map of an input video. If the input video is an abnormal one, the reconstruction error is relatively large; otherwise, the reconstruction error is small. Following this idea, Sabokrou *et al.* [15] proposed an adversarially learned one-class (ALOCC) reconstruction model. This approach trains an Encoder-Decoder network to reconstruct the feature map of an input video for accident/abnormality detection. Some other approaches [24]–[26] have been proposed based on the Generative Adversarial Networks (GANs). Zheng *et al.* [26] proposed one-class adversarial nets (OCAN), which use the LSTM-Autoencoder to generate the feature map of an input video for accident/abnormality detection. In addition, some similar approaches have also been proposed, such as Fence Generative Adversarial Networks (FGAN) [27] and Single-Objective Generative Adversarial Active Learning (SOGAAL) [28].

Although the video level approaches can effectively solve the uneven distribution of samples in the dataset. However, as the traffic accidents have a short duration in driving videos, the accident/abnormality detection suffers from the long-tailed distribution problem. Moreover, the backgrounds of driving

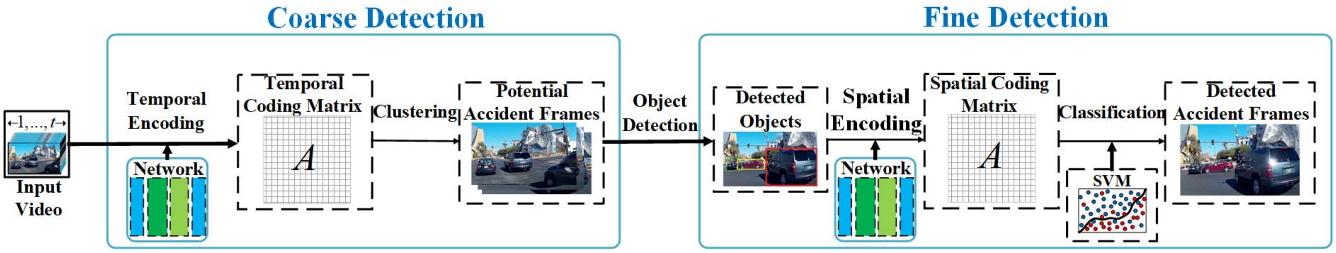

Fig. 1. The framework of the proposed approach, which mainly includes the temporal encoding for coarse detection and the spatial encoding for fine detection.

videos are dynamic and complex. Therefore, it is hard for these video-level detection approaches to achieve desirable performance for accident/abnormality detection.

B. Segment-Level Detection Approaches

To address the problem of long-tailed distribution, some segment-level accident/abnormality detection approaches have been proposed. Motivated by the technique of object detection in images such as Faster Region-CNN (Faster R-CNN) [29], the segment-level detection approaches are usually the two-stage framework. First, a set of proposals are segmented from each video, and these segment proposals are fed into a pre-trained classifier to detect the accidents/abnormalities.

Sultani *et al.* [17] adopted the 3D Convolutional Networks (C3D) with the multiple-instance learning (MIL) technique to build the framework of segment-level accident/abnormality detection. This approach divided each video into 32 segments and treated the videos as bags and segments as instances. Then, it inputs these segments into the 3D Convolutional Networks pretrained by the MIL technique to extract the segment features, and then fed these segment features into fully connected layers of C3D to obtain the anomaly scores. Finally, the anomaly segments are determined by their anomaly scores. Instead of extracting C3D features, Zhu and Newsam [18] computed the optical flows of the segments as the segment features and adopted the attention mechanism with the MIL technique to improve the detection accuracy. Similarly, Lin *et al.* [30] proposed a segment-level accident/abnormality detection approach based on the dual-branch network, which consists of spatial-temporal dynamic subnetwork and the interactive dynamic subnetwork. The input of the spatial-temporal dynamic network is the original video segments, while the input of the interactive dynamic network is the feature maps that capture the interactive information between pedestrians and surrounding environments. Then, the two networks are trained simultaneously by the principle of MIL for accident/abnormality detection. In addition, Landi *et al.* [31] further considered the locality of the occurrence of anomalies for segment-level accident/abnormality detection.

Segment-level approaches can localize and detect accident/abnormality segments in complex long videos. However, if there is more than one abnormality instance in a long and complex video, these segment-level approaches are inefficient for accident/abnormality detection in the VANET environment.

C. Frame-Level Detection Approaches

Generally, the frame-level approaches constructed the current frames using a variety of image generation models with the input of the previous frames. Then, the construction errors are computed by comparing the constructed current frame and the original current frame for accident/abnormality detection. These approaches can be classified into two categories, *i.e.*, the GAN network-based and Autoencoder-based approaches.

In the early work, to detect pedestrian violations of traffic rules in crowded scenes, Liu *et al.* [32] proposed a frame-level detection approach based on generator-discriminator structure. They used the U-Net network as the generator to predict the current frame and used a discriminator network to determine whether or not the predicted frame is an abnormal one. Subsequently, with the development of the GANs, GAN-based anomaly detection has become one of the most popular frame-level anomaly detection techniques. Schlegl *et al.* [19] proposed the Anomaly Detection with Generative Adversarial Networks (AnoGAN) to learn the distribution of normal frames to generate the current frame. Then, the generated current frame and the original current frame were compared to determine whether the current frame is an abnormal one. However, the training of AnoGAN is quite time-consuming. To solve the computational inefficiency problem of AnoGAN, some other GAN-based approaches have been proposed, such as Efficient Generative Adversarial Networks-Based (EBGAN), fast Anomaly Detection with Generative Adversarial Networks (fast AnoGAN) [33] and Bi-directional Generative Adversarial Networks (BiGAN) [34]. Those GANs have improved the accuracy of anomaly detection because of the superior capability of generating realistic frames. However, in most cases, training a GAN-based anomaly detection model is very difficult, mainly due to the failure converge and mode collapse. To address these problems, some autoencoder-based approaches have been proposed. Hasan *et al.* [35] proposed an end-to-end framework based on a fully convolutional feed-forward autoencoder. The current frame was generated by the autoencoder with the input of the previous frames. Similarly, Medel and Savakis [36] proposed Convolutional Long Short-Term Memory Networks (ConvLSTM) for frame-level accident/abnormality detection.

In these frame-level detection approaches, since the frame construction process should be conducted many times, they suffer from the problem of high computational complexity. Therefore, it is still not feasible to directly adopt these approaches for accident/abnormality detection in VANET environment, since real-time detection is usually required.

III. The Proposed Approach

In this section, the proposed approach is elaborated. In Section III-A, the framework of the proposed approach is first introduced. In Section III-B, temporal features of frames are encoded for coarse detection. In Section III-C, spatial features of objects detected from frames are encoded for fine detection.

A. The Framework of Proposed Approach

The framework of the proposed approach is based on coarse-to-fine detection using spatial-temporal feature encoding with a multilayer neural network [21], as shown in Fig. 1. It consists of two main stages: temporal feature encoding for coarse detection and spatial feature encoding for fine detection.

1) In the stage of coarse detection, for a given traffic video, we first encode the two temporal features, *i.e.*, the HOF features [22] and the ordinal features of frames as the feature map by the multilayer neural network to cluster the frames. From the frame clusters, we detect the potential accident frames.

2) In the stage of fine detection, we encode the CNN features and spatial relationships of the objects detected from the potential accident frames through the multilayer neural network. The encoded spatial features are fed into a trained SVM to confirm whether there is an accident happen in the potential accident frames. Consequently, we can localize and detect the accident frames in the given video.

According to the literature [21], the multilayer neural network provides a representation learning platform with unsupervised/supervised and compressed/sparse learning. Thus, we employed the multilayer neural network for encoding the features of frames and objects.

B. Temporal Feature Encoding for Coarse Detection

In the stage of coarse detection, we extract and encode temporal features from a given video as a coding matrix. Then, based on the coding matrix, we cluster the video frames. Finally, from the frame clusters, we coarsely detect the potential accident frames. Thus, there are three main steps: temporal feature encoding, frame clustering, and potential accident frame detection.

1) Temporal Feature Encoding: Given a video with t frames, we first extract the motion feature vectors, *i.e.*, the histogram of optical flow (HOF) descriptor [22], which represents the temporal trajectory information of these frames. According to [22], the HOF descriptor is extracted from each frame and its 15 consecutive frames and thus the whole video is presented as $V = \{v_1, v_2, \ldots, v_m\}$, where $m = t - 15$. Then, the feature vector sequence and the temporal ordinal information of the video frames are encoded for the coding matrix generated by a multilayer neural network. In coding matrix generation, we label the feature vectors of video frames $V = \{v_1, v_2, \ldots, v_m\}$ according to the indices of frames denoted as $L = \{l_1, l_2, \ldots, l_m\}$. Then, the feature vectors and their labels are fed into the multilayer neural network, and then the parameters of the model are used as a coding matrix for frame clustering. The following five steps are performed to generate the coding matrix A, as shown in Fig. 2.

Algorithm 1 Frame Clustering Using Self-Representation Constrained Low-Rank Representation (SRLRR)

```
Input:
Coding Matrix $A = [a_1, a_2, \ldots, a_m] \in \mathbb{R}^{d \times m}$, parameters $\beta, \lambda$;
Output:
The coefficient matrix $Z^*$ and noise $E^*$;
Initialize the parameters. $Y_1^0, Y_2^0, Y_3^0, Y_4^0 = 0$,
    $\mu^0 = 10^{-8}, \mu_{max} = 10^{30}, \rho = 1.1, \varepsilon = 10^{-8}, k = 0$
Initialize the variables. $Z^0, J^0, T^0 \in \mathbb{R}^{m \times m}$ and
    $E^0 \in \mathbb{R}^{d \times m}$
while ($\|A - AZ^k - E^k\|_\infty > \varepsilon$ and $\|Z^k - J^k\|_\infty > \varepsilon$ and
    $\|Z^k - T^k\|_\infty > \varepsilon$ and $\|1_n Z^k - 1_n\|_\infty > \varepsilon$)
    do:
    Update $J^k$;
    Update $T^k$;
    Update $Z^k$;
    Update $E^k$;
    Update $Y_1^k, Y_2^k, Y_3^k, Y_4^k, \mu^k$;
end while
return the coefficient matrices $Z^* = Z^k, E^* = E^k$
Build an affinity graph using Eq. (14)
Use NCut algorithm to generate clusters
```

Step 1: We use the vectors $\{(v_k, l_k)\}_{k=1}^m$, $v_k \in \mathbb{R}^D$, $l_k \in \mathbb{R}^m$ to represent the training data, and every hidden node is formed by several sub-nodes. The parameters of sub-nodes and the coding matrix are initialized randomly as

$$\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}} = \mathrm{g}\left(\hat{w}_c^j \cdot \boldsymbol{V} + \hat{b}_c^j\right), \quad \left(\hat{w}_c^j\right)^T = \left(\hat{w}_c^j\right)^{-1}, \quad \left\|\hat{b}_c^j\right\| = 1 \tag{1}$$

where $\hat{\boldsymbol{w}}_{\boldsymbol{c}}^{\boldsymbol{j}} \in \mathbb{R}^{\boldsymbol{D} \times \boldsymbol{d}}$ and $\hat{\boldsymbol{b}}_{\boldsymbol{c}}^{\boldsymbol{j}} \in \mathbb{R}$ are weights and bias of nodes in the coding layer, respectively, d represents the number of sub-network nodes in each generally hidden node in the coding layer, and D is the dimensions of video features.

Step 2: We use a sigmoid or sine function as the activation function $\mathbf{g}(\cdot)$ for continuous desired output $L = \{l_1, l_2, \ldots, l_m\}$, the weights in the learning layer are computed as

$$\hat{w}_l = u_n(L) \cdot \left(\mathrm{g}^{-1}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}\right)\right)^T \left((C/I) + \mathrm{g}^{-1}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}\right)\left(\mathrm{g}^{-1}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}\right)\right)^T\right)^{-1}$$
$$\hat{w}_l \in \mathbb{R}^{m \times d} \tag{2}$$

where C is a constant value, u_n is a normalized function while $u_n(L) : \mathbb{R} \to (0\ ,\ 1]$, and $\mathbf{g}^{-1}(\cdot)$ represents the inverse function of $\mathbf{g}(\cdot)$. The formulation is given as follows:

$$\mathrm{g}^{-1}(\cdot) = \begin{cases} arcsin(\cdot) & if\ \ \mathrm{g}(\cdot) = sin(\cdot) \\ -log\left(\frac{1}{(\cdot)} - 1\right) & if\ \ \mathrm{g}(\cdot) = 1/\left(1 + e^{(\cdot)}\right) \end{cases} \tag{3}$$

The bias in the learning layer is computed as

$$\hat{b}_l = \sqrt{mse\left(\hat{\boldsymbol{w}}_{\boldsymbol{l}}^{\boldsymbol{j}} \cdot \mathrm{g}^{-1}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}\right) - u_n(L)\right)}, \quad \hat{b}_l \in \mathbb{R} \tag{4}$$

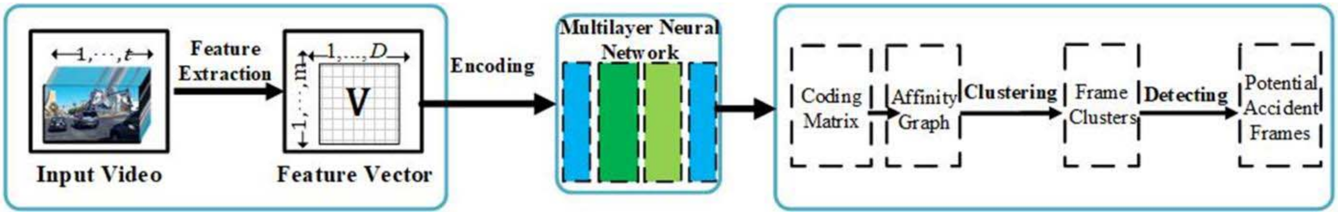

Fig. 2. The illustration of temporal feature encoding for coarse detection.

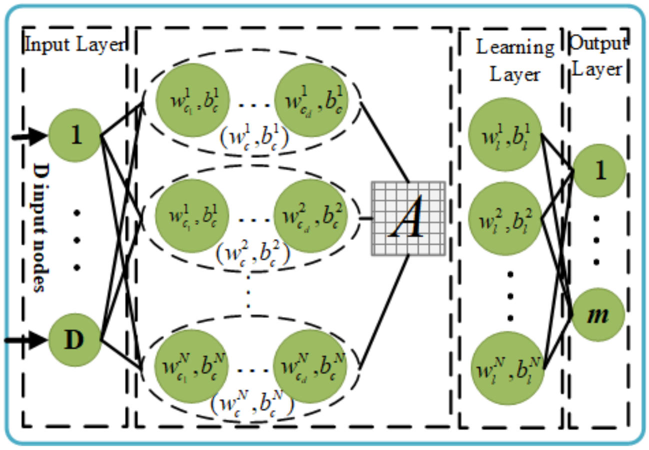

Fig. 3. The structure of multilayer neural network [21].

Step 3: Calculate the residual error e_j and error feedback data P_j as

$$e_j = L - u_n^{-1} \mathrm{g}\left(\hat{\boldsymbol{w}}_{\boldsymbol{l}} \cdot \boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}} + \hat{\boldsymbol{b}}_{\boldsymbol{l}}\right) \quad (5)$$

$$P_j = e_j \cdot \left(\hat{\boldsymbol{w}}_{\boldsymbol{l}}^{\boldsymbol{T}} \left(C/I + \hat{\boldsymbol{w}}_{\boldsymbol{l}} \left(\hat{\boldsymbol{w}}_{\boldsymbol{l}}\right)^{\boldsymbol{T}}\right)^{-1}\right) \quad (6)$$

$$P_j = u_n \left(P_j + \mathrm{g}^{-1}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}\right)\right) \quad (7)$$

where $u_n^{-1}(\cdot)$ represents the inverse function of $u_n(\cdot)$.

Step 4: In the output layers, the multilayer neural network aims to obtain the smallest training error and smallest output weights by minimizing:

$$\left\|\hat{w}_o^i\right\|_p^{u_1} + C\left\|\sum_{i=1}^{D} \hat{w}_o^i \mathrm{g}\left(\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}}, \hat{\boldsymbol{w}}_{\boldsymbol{l}}, \hat{\boldsymbol{b}}_{\boldsymbol{l}}\right) - L\right\|_q^{u_2}, \quad i = 1, \ldots, m \quad (8)$$

where $u_1 > 0, u_2 > 0, p, q = \{0, \frac{1}{2}, 1, 2, \ldots -, +\infty\}$, and C is a constant value. In the output layer, the weight $\hat{w}_o^i$ is computed by

$$\hat{w}_o^i = \frac{\left\langle e_{i-1}, \mathrm{g}(A_c^i, w_o^r, b_o^r)\right\rangle}{\left\|\mathrm{g}\left(A_{\boldsymbol{c}}^{\boldsymbol{i}}, w_o^r, b_o^r\right)\right\|^2} \quad (9)$$

where $\langle \cdot, \cdot \rangle$ means the Moore-Penrose function, and $\left(w_o^r, b_o^r\right)$ represents the parameters between hidden nodes and output nodes.

Step 5: Set $j = j + 1$, and then add a new general node $\hat{w}_c^j, \hat{b}_c^j$ in the coding layer as

$$\hat{w}_c^j = \mathrm{g}^{-1}\left(u_n\left(P_{j-1}\right)\right) \cdot V^{-1}, \quad \hat{w}_c^j \in \mathbb{R}^{D \times d} \quad (10)$$

$$\hat{b}_c^j = \sqrt{mse\left(\hat{w}_c^j \cdot V - P_{j-1}\right)}, \quad \hat{b}_c^j \in \mathbb{R} \quad (11)$$

Subsequently, update the coding matrix by

$$\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{i}} = \sum_{S=1}^{j} u_S^{-1} \mathrm{g}\left(V, \hat{w}_c^S, \hat{b}_c^S\right) \quad (12)$$

Step 6: Repeat **Step 2 to Step 5** $N - 1$ times where N is the number of general nodes in the coding layer. The parameters $\left\{\hat{a}_c^j, \hat{b}_c^j\right\}_{j=1}^{N}$ are the optimal projecting parameters of the coding matrix in the coding layer. In this paper, N is equal to the number of feature vectors m. Finally, we can obtain the coding matrix of temporal features $\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{N}}$ by

$$\boldsymbol{A}_{\boldsymbol{c}}^{\boldsymbol{N}} = \sum_{\boldsymbol{w}=1}^{\boldsymbol{N}} u_w^{-1} \mathrm{g}\left(V, \hat{w}_c^w, \hat{b}_c^w\right) = \boldsymbol{A}^* \quad (13)$$

2) Frame Clustering: Then, to facilitate the coarse detection, we try to cluster the video frames according to the obtained coding matrix.

Subspace clustering has been adopted in many real-world applications such as motion segmentation [37], face recognition [38], and image retrieval [39]. Recently, graph-based clustering algorithms have been proposed to obtain low-rank coefficient matrices for clustering. The most typical algorithms are Low-Rank Representation (LRR) [40] and Sparse Subspace Clustering (SSC) [41]. LRR has been proven to be superior to SSC. Thus, many extended versions of LRR have been proposed such as self-representation constrained low-rank representation (SRLRR) [42]. In these approaches, the obtained low-rank coefficient matrices are essentially affinity matrices, in which the similarities between homogeneous data samples are enhanced while the similarities between heterogeneous data samples are decreased. Thus, the low-rank coefficient matrix will be more suitable for revealing the subspace relationship of data and clustering the data. Therefore, in the proposed approach, we adopt SRLRR for the obtained coding matrix to compute the low-rank coefficient matrix to cluster video frames.

The SRLRR algorithm aims to compute a coefficient matrix with minimal rank, denoted as Z, which satisfies $A = AZ + E$ and $Z = ZZ = Z^2$. Where A represents the input matrix and E represents the residual between coefficient matrix Z and input matrix A. The optimal solution by the SRLRR is a block-diagonal matrix. In the proposed approach, the algorithm of frame clustering using SRLRR is summarized in Algorithm 1. Reference [42] consequently, an affinity graph G can be obtained by

$$G_{(i,j)} = \frac{\left(\left|Z_{(i,j)}\right| + \left|Z_{(j,i)}\right|\right)}{2} \quad (14)$$

$G_{(i,j)}$ and $Z_{(i,j)}$ are the (i, j)-th elements of G and Z, respectively.

3) Potential Accident Frame Detection: In the affinity graph, if two feature vectors are highly correlated, they tend to be in the same diagonal block. Then a spectral clustering algorithm, *i.e.*, normalize cut (Ncut) [43], is employed to produce the final segmentation results and the indices of the potential accident frames. Specifically, the border frames in the clusters are detected by the Ncut algorithm as the potential accident frames.

C. Spatial Feature Encoding for Fine Detection

On the fine detection stage, we encode the spatial features of the objects detected from the potential accident frames, including the CNN features of the objects generated by Faster-RCNN and the spatial relationships captured among these objects to compute the spatial coding matrix. Then, we feed the spatial coding matrix into trained SVM to confirm the potential accident frames for fine detection. Thus, there are three main steps: object detection, spatial feature encoding, and accident frame confirmation.

1) Object Detection: After the coarse detection, the potential accident frames have been obtained. Since the spatial features of frames should be captured, it is necessary to detect objects from the obtained potential accident frames.

In the literature, many object detection networks have been proposed, such as YOLO [44] and Faster-RCNN [29]. In the proposed approach, we adopt Faster-RCNN to detect the objects from the potential accident frames since it has high efficiency with desirable accuracy in object detection. The Faster-RCNN consists of four parts: Convolution layers, Region Proposal Network (RPN), RoIAlign (Roi) Pooling, and Classification.

From the last convolution layer of Faster-RCNN [29], we can get the feature vector $F \in \mathbb{R}^N$ of a single video frame, where N means the dimension of the feature vector. Then the feature vector is fed into the RPN to generate a set of object proposals. As the DoTA dataset used in the experiments has provided us with ground-truth boxes of abnormal objects in the frames, we can assign the positives and negatives according to the overlapping area between anchor and ground-truth. The loss function of training is defined as:

$$L\left(\{p_i\}, \{b_i\}\right) = \frac{1}{N_{cls}} \sum_i L_{cls}\left(p_i, p_i^*\right) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}\left(b_i, b_i^*\right) \tag{15}$$

where i is the index of anchor, p_i is the predicted probability of anchor i being an object, p_i^* is the ground-truth label, b_i is a vector of 4 parameters representing the predicted bounding box, b_i^* is a vector of 4 parameters representing the ground-truth box associated with a positive anchor, L_{cls} is the log loss over two classes, and L_{reg} is the regression loss.

Consequently, we can obtain a set of objects detected from each potential accident frame. Each object can be represented by four parameters (x, y, w, h). Where, x, y are the coordinates of the center point of the object bounding box, and w and h are the weight and height of the object bounding box, respectively.

2) Spatial Feature Encoding: In this step, we encode the CNN features and the spatial relationships of objects detected from each potential accident frame, where the CNN features are the convolutional features extracted from the last convolutional layer of Faster-RCNN, and the spatial relationships of objects are the spatial relative positions among these objects and the relative size relationship among these objects. We separately encode the two spatial relationships with the CNN features of objects to generate two corresponding coding matrixes, as shown in Fig. 4. More details are given as follows.

The CNN feature vectors of objects are represented by $I = \{I_1, I_2, \ldots, I_n\}$, where n is the number of objects in the frame, and the bounding boxes of each object are represented by $\{(x_i, y_i, w_i, h_i)\}_i^n$. Then, we label the feature vectors of the target boxes based on the order of the target boxes along the x-axis positive direction as $o = \{1, 2, \ldots, n\}$. Next, the feature vectors I and their corresponding order labels o are fed into the multilayer neural network. By the same steps, *i.e.*, Step 1 to Step 5 in Section III-B, a new coding matrix A_1 can be obtained.

Similarly, for $\{(x_i, y_i, w_i, h_i)\}_i^n$, we also label the feature vectors of the target boxes based on the descending order of the sizes of target boxes. Next, the feature vectors $I = \{I_1, I_2, \ldots, I_n\}$ and their corresponding order labels are input into a multilayer neural network to generate another coding matrix A_2.

3) Accident Frame Confirmation: In natural driving videos, the recognition of traffic accidents is a typical problem of multi-classification. Thus, we employ a trained SVM classifier with the input of coding matrices A_1 and A_2 in the proposed approach. In this step, we concatenate A_1 and A_2 to form a single vector and then feed it into the SVM. The DoTA dataset used in the experiments has provided detailed annotation of accident events, and the objective of SVM is to find a hyperplane that can distinguish the samples of input feature vectors. The hyperplane can be defined by

$$g\left(x\right) = w^T x + b \tag{16}$$

The geometric distance from the sample points to the hyperplane is computed by

$$d_i = y_i\left(\frac{|wx_i + b|}{\|w\|}\right) \tag{17}$$

where d_i is the geometric distance of i-th sample to the hyperplane, and y_i is the label of i-th sample. The maximum partitioned hyperplane problem of the SVM model can be solved by the following constrained optimization problem:

$$min_{(w,b)} \frac{1}{2} \|w\|^2 \tag{18}$$

As a result, by the SVM, we can finally obtain the detected accident frames from a given traffic video.

IV. Experiments and Analysis

A. Datasets

There are some publicly available video datasets for traffic/abnormality detection. However, to the best of our knowledge, most of those traffic datasets are surveillance videos

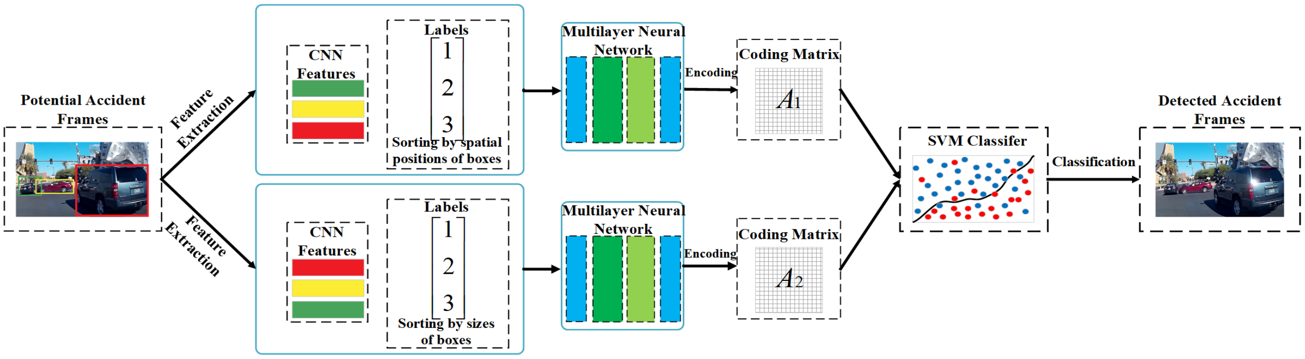

Fig. 4. The illustration of spatial feature encoding for coarse detection.

TABLE I

COMPARISON BETWEEN DIFFERENT TRAFFIC ACCIDENT DATASETS

Dataset	Videos	Frames	Annotations
StreetAccident	620	62000(20fps)	temporal
A3D	1500	128175(10fps)	temporal
DATA	2000	648476(30fps)	temporal, spatial (eye-gaze)
DoTA	4677	731932(10fps)	temporal, spatial (tracklets), categories

TABLE II

TRAFFIC ACCIDENT CATEGORIES IN THE DOTA DATASET

ID	Short	Anomaly Categories
1	ST	Collision with another vehicle which starts, stops, or is stationary
2	AH	Collision with another vehicle moving ahead or waiting
3	LA	Collision with another vehicle moving laterally in the same direction
4	OC	Collision with another oncoming vehicle
5	TC	Collision with another vehicle which turns into or crosses a road
6	VP	Collision between vehicle and pedestrian
7	VO	Collision with an obstacle in the roadway
8	OO	Out-of-control and leaving the roadway to the left or right
9	UK	Unknown

captured by the cameras on the road. And in the VANET environment, the videos are usually the driving videos captured by the vehicle-mounted cameras for a variety of tasks such as road safety warning [9], autonomous driving [11], traffic flow management [12], and pedestrian anomaly detection [45]. Different from the surveillance videos, in which the backgrounds are usually static and simple, the backgrounds of driving videos are dynamic and complex.

In [46], a new publicly available driving dataset, named DoTA, is introduced for traffic accident detection and recognition. It consists of more than 6000 video clips with the resolution of 1280 × 720 derived from YouTube, which is captured from different countries under different weather and lighting conditions. Table I compares DoTA with other traffic video datasets. In DoTA, there are a lot of temporal, spatial, and categorical annotations of accidents for frames and objects in the videos.

Each video is labeled with a set of the start and end times, which divides the video into three parts: normal frames preceding the accident, the accident frames, and post-accident frames. DoTA is the first traffic video dataset that provides detailed spatio–temporal annotations of anomalous objects. The dataset has provided each anomalous object with a unique track ID, and the label of their bounding box. Moreover, each video is assigned by one of 9 categories listed in Table II. Thus, we adopt this dataset in the experiments to evaluate the performances of different approaches.

B. Evaluation of Video Clustering Algorithm

In this section, we evaluate the accuracy of the proposed approach for traffic accident detection. The dataset has provided the ground-truth classification labels of the individual frames in the video samples. Then, we use the pair-counting measurement to evaluate the accuracy of the clustering approaches. The employed pair-counting measurement includes two major variables (p_1, p_2). The p_1 refers to the percentage of both the ground-truth and clustered frames are assigned to same clusters. The p_2 refers to the percentage of both the ground-truth and clustered frames are assigned to different clusters. Furthermore, the average value of p_1 and p_2 indicates the clustering performance.

There are three key parameters related to the performance of the video clustering algorithm. In the step of temporal feature encoding, the parameter C in Eq. (8) has an impact on the generation of diagonal AA^T and the frame clustering results. Thus, we test a set of values for parameter C, *i.e.*, $C \in \left\{2^{-4}, \ldots, 2^{-8}\right\}$. Fig. 5 shows the performance of video frame clustering by using different parameters C on the DoTA dataset. From Fig. 5, it is clear that the clustering algorithm achieves the best performance when $C = 8$. Also, we test the impacts of parameters λ and β used in the SRLRR algorithm on the clustering performance, when λ and β vary in the range of [0.001,7] and [0.01,10], respectively, as shown in Fig. 6. From Fig. 6, $\lambda = 5$ and $\beta = 10$ provide the optimal performance in frame clustering. Thus, we set $C = 5, \lambda = 5$, and $\beta = 10$ in the following experiments.

Moreover, we compare the performance of our clustering algorithm with the three other clustering algorithms. In all

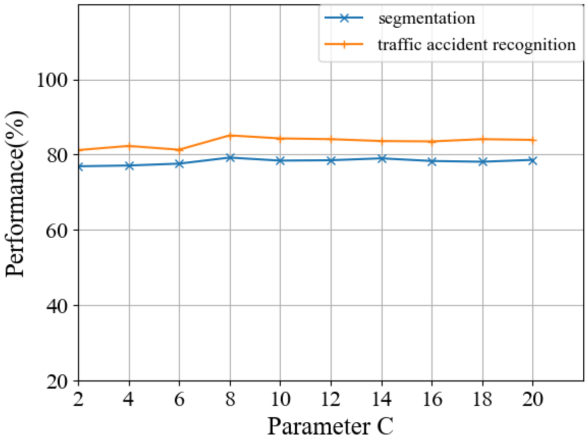

Fig. 5. Evaluation of parameter C for video action clustering.

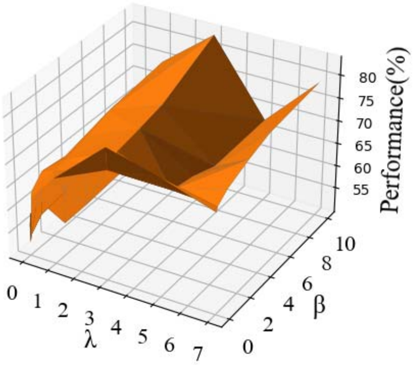

Fig. 6. Impacts of parameter λ and β on video clips clustering.

TABLE III
COMPARISON OF CLUSTERING ALGORITHMS

Dataset	Approaches	Accuracy
DoTA	SSC	41.8%±3.2%
	LSR	47.2%±2.5%
	OSC	70.1%±3.1%
	Our Approach	85.3%±1.8%

of those clustering algorithms, the multilayer neural network is used to encode the temporal features, and then a clustering algorithm is used to cluster frames. The difference among them is that our clustering algorithm is based on LRRSR, while the three others use SSC, LSR, and Ordinal Subspace Clustering (OSC), respectively.

Table III lists the accuracy of our clustering algorithm and the state-of-the-art on the DoTA dataset. It is clear that our clustering algorithm achieves high accuracy, and is superior to the three others for the following reasons. Since the temporal features including the trajectories and ordinal information of frames are encoded, the generated code matrix can sufficiently capture the relationships among the frames. Moreover, the powerful SRLRR algorithm can refine the code matrix to reflect the relationships among frames. As a result, the potential traffic accident frames can be easily obtained in the affinity graph G. That is beneficial for the final accident detection performance.

TABLE IV
THE AUC OF DIFFERENT APPROACHES ON THE DOTA DATASET

Approaches	Type	Input	AUC
ConvAE [35]	Unsupervised	Gray	64.3
ConvAE		Flow	66.3
ConvLSTMAE [36]		Gray	53.8
ConvLSTMAE		Flow	62.5
AnoPred [33]		RGB	67.5
AnoPred + Mask [46]		Masked RGB	64.8
TAD [47]		Box+ Flow	69.2
TAD + ML [46]		Box+ Flow	69.7
Ensemble [46]		RGB+Box+ Flow	73.0
FC [46]	Supervised	RGB	61.7
LSTM [48]			63.7
Encoder-Decoder [49]			73.6
TRN [50]			78.0
Our Approach		Flow+RGB	79.3

C. Performance Evaluation and Comparison

In this section, we compare the accuracy of the proposed approach with those of the 11 state-of-the-art approaches, which are denoted as Convolutional AutoEncoder (ConvAE) [35], Convolutional Long Short-Trem Memory AutoEncoder (ConvLSTMAE) [36], AnoPred [33], AnoPred+Mask [46], Fully Connected (FC) [46], Traffic Accident Detection (TAD) [47], TAD+Margin Learning (ML) [46], Ensemble [46], LSTM [48], Encoder-Decoder [49], and Temporal Recurrent Networks (TRN) [50]. Where, FC is the video-level detection approach, and TAD, TAD+ML and Ensemble are the segment-level detection approaches based on localization, while ConvAE, ConvLSTMAE, AnoPred, AnoPred + Mask, LSTM, and Encoder-Decoder are the frame-level detection approaches based on construction error.

Table IV shows the detection accuracy of different approaches on the DoTA dataset. In Table IV, "Gray, Flow, RGB" mean that the Gray images, Gradient histogram, and RGB images are used as the inputs in different approaches, while "Masked RGB and Box" mean the RGB images with masks and the bounding box are used as the inputs, respectively. As shown in Table IV, it is clear that the performance of approaches with optical flow input is better than that with grayscale input. Moreover, supervised approaches such as FC, LSTM, Encoder-Decoder generally achieve higher Area Under Curve (AUC) than unsupervised approaches. That is because the proposed approach combines the advantages of optical flow input and supervised learning manner, and thus it achieves the highest AUC.

Table V shows the detection results of supervised approaches, *i.e.*, FC, LSTM, Encoder-Decoder, and TRN for each class. It can be clearly observed that TRN achieves higher AUC only on the classes of LA, TC, AH*, and OC*, while the proposed approach achieves higher AUCs on more classes including ST, AH, OC, VP, ST*, LA*, TC*, VO*, and OO*. Table V also shows that detecting the classes including LA, VP, AH*, LA*, and OO* is challenging for all approaches. However, it is clear that our approach still achieves the highest accuracy, and is superior to the three others.

Table VI also lists the number of parameters of different networks including AlexNet, VGG16, ResNet50, GoogleNet

TABLE V
DETECTION ACCURACY FOR EACH INDIVIDUAL ACCIDENT CATEGORY (AUC)

Approaches	ST	AH	LA	OC	TC	VP	ST*	AH*	LA*	OC*	TC*	VO*	OO*
FC	69.9	73.6	75.2	69.7	73.5	66.3	70.9	62.6	60.1	65.6	65.4	64.9	57.8
LSTM	66.3	72.2	64.2	65.4	65.6	66.6	72.9	63.7	60.6	66.9	65.7	64.0	59.9
Encoder-Decoder	67.3	77.4	71.1	68.6	69.2	65.1	75.1	66.2	66.8	74.1	72.0	69.7	69.2
TRN	73.3	81.2	**74.0**	73.4	**75.1**	70.1	77.5	**69.8**	68.1	**76.7**	73.9	71.2	69.6
Our Approach	**75.2**	**84.5**	72.1	**77.3**	72.8	**71.9**	**80.6**	65.6	**69.9**	76.5	**74.2**	**75.6**	**70.5**

TABLE VI
COMPARISON OF THE NUMBER OF PARAMETERS AND COMPUTATION TIME

Approaches	Number of parameters	Training time	Average detection time
AlexNet	62.3M	5.1h	0.59s
VGG16	138M	15.4h	1.108s
ResNet50	6.7M	3.5h	0.526s
GoogleNet	6.8M	1.6h	0.295s
Our Approach	6.6M	0.92h	0.201s

and the training time on the DoTA dataset. It is clear that the training time of the proposed approach is much less than that of the three others. The average detection time of the proposed approach is only about 0.201s, mainly because of efficient feature encoding and frame clustering algorithm in coarse detection stage. Therefore, the proposed approach can meet the real-time detection requirement in VANET environment.

V. CONCLUSION

This paper has presented an accident detection approach to detect the accident frames from traffic videos in the VANET environment. We proposed a coarse-to-fine detection framework based on the spatial-temporal feature encoding with the multilayer neural network. The proposed approach has solved the main challenges of abnormality/accident detection, including the problems of long-tailed distribution and heterogeneous anomaly classes. The experiment results demonstrate the proposed approach not only achieves higher detection accuracy than the state-of-the-arts, but also needs less training time and detection time. Consequently, the requirement of abnormality/accident detection can be satisfied in the VANET environment.

Moreover, it is notable that the existing accident/ abnormality detection approaches usually ignore the detection of pedestrian abnormality, which is very helpful for the persons with impaired vision and the elderly groups. In future, we will extend the proposed approach with the help of mobile devices for pedestrian abnormality detection. Specifically, the locations, statuses, and surrounding environments of pedestrians can be detected by the Global Positioning System (GPS), sensors, and cameras of their mobile devices, respectively, and these data is then utilized to detect pedestrian abnormalities.

REFERENCES

[1] B. B. Gupta, A. Gaurav, C.-H. Hsu, and B. Jiao, "Identity-based authentication mechanism for secure information sharing in the maritime transport system," *IEEE Trans. Intell. Transp. Syst.*, early access, Nov. 15, 2021, doi: 10.1109/TITS.2021.3125402.

[2] Z. Zhou, A. Gaurav, B. B. Gupta, M. D. Lytras, and I. Razzak, "A fine-grained access control and security approach for intelligent vehicular transport in 6G communication system," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 2, 2021, doi: 10.1109/TITS.2021.3106825.

[3] M. Boubenia, A. Belkhir, and F. M. Bouyakoub, "Combining linked open data similarity and relatedness for cross OSN recommendation," *Int. J. Semantic Web Inf. Syst.*, vol. 16, no. 2, pp. 59–90, Apr. 2020.

[4] F. Lyu *et al.*, "MoMAC: Mobility-aware and collision-avoidance MAC for safety applications in VANETs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10590–10602, Nov. 2018.

[5] M. M. Hamdi, L. Audah, S. A. Rashid, A. H. Mohammed, S. Alani, and A. S. Mustafa, "A review of applications, characteristics and challenges in vehicular ad hoc networks (VANETs)," in *Proc. Inter. Congr. Hum.-Comput. Intera, Optim. Rob. Applic (HORA)*, Jun. 2020, pp. 1–7.

[6] M. M. Hamdi, L. Audah, S. A. Rashid, and M. A. Al-shareeda, "Techniques of early incident detection and traffic monitoring centre in VANETs: A review," *J. Commun.*, vol. 15, no. 12, pp. 896–904, 2020.

[7] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, "Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4757–4769, Jul. 2021.

[8] F. Mirsadeghi, M. K. Rafsanjani, and B. B. Gupta, "A trust infrastructure based authentication method for clustered vehicular ad hoc networks," *Peer-Peer Netw. Appl.*, vol. 14, no. 4, pp. 2537–2553, 2020.

[9] Y. Chen and A. Boukerche, "A novel lane departure warning system for improving road safety," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[10] A. Tewari and B. B. Gupta, "Secure timestamp-based mutual authentication protocol for IoT devices using RFID tags," *Int. J. Semantic Web Inf. Syst.*, vol. 16, no. 3, pp. 20–34, Jul. 2020.

[11] F. Codevilla, E. Santana, A. Lopez, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9329–9338.

[12] N. Ivanov, F. Netjasov, R. Jovanović, S. Starita, and A. Strauss, "Air traffic flow management slot allocation to minimize propagated delay and improve airport slot adherence," *Transp. Res. A, Policy Pract.*, vol. 95, pp. 183–197, Jan. 2017.

[13] D. Peraković, M. Periša, and V. Remenar, "Model of guidance for visually impaired persons in the traffic network," *Transp. Res. F, Traffic Psychol. Behav.*, vol. 31, pp. 1–11, May 2015.

[14] M. Periša, I. Cvitić, D. Peraković, and S. Husnjak, "Beacon technology for real-time informing the traffic network users about the environment," *Transport*, vol. 34, no. 3, pp. 373–382, Jun. 2019.

[15] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli, "Adversarially learned one-class classifier for novelty detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3379–3388.

[16] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, no. 1, pp. 45–66, Jan. 2004.

[17] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.

[18] Y. Zhu and S. Newsam, "Motion-aware feature for improved video anomaly detection," 2019, *arXiv:1907.10211*.

[19] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.* Cham, Switzerland: Springer, Jun. 2017, pp. 146–157.

[20] H. Zenati, C. Sheng Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, "Efficient GAN-based anomaly detection," 2018, *arXiv:1802.06222*.

[21] Y. Yang and Q. M. J. Wu, "Multilayer extreme learning machine with subnetwork nodes for representation learning," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2570–2583, Nov. 2016.

[22] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3551–3558.

[23] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015, *arXiv:1511.08458*.

[24] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[25] K. T. Chui, R. W. Liu, M. Zhao, and P. O. D. Pablos, "Predicting students' performance with school and family tutoring using generative adversarial network-based deep support vector machine," *IEEE Access*, vol. 8, pp. 86745–86752, 2020.

[26] P. Zheng, S. Yuan, X. Wu, J. Li, and A. Lu, "One-class adversarial nets for fraud detection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jul. 2019, pp. 1286–1293.

[27] P. C. Ngo, A. A. Winarto, C. K. L. Kou, S. Park, F. Akram, and H. K. Lee, "Fence GAN: Towards better anomaly detection," in *Proc. IEEE 31st Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2019, pp. 141–148.

[28] Y. Liu, Z. Li, C. Zhou, Y. Jiang, J. Sun, M. Wang, and X. He, "Generative adversarial active learning for unsupervised outlier detection," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 8, pp. 1517–1528, Aug. 2020.

[29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 91–99.

[30] S. Lin, H. Yang, X. Tang, T. Shi, and L. Chen, "Social MIL: Interaction-aware for crowd anomaly detection," in *Proc. 16th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2019, pp. 1–8.

[31] F. Landi, C. G. M. Snoek, and R. Cucchiara, "Anomaly locality in video surveillance," 2019, *arXiv:1901.10364*.

[32] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection—A new baseline," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6536–6545.

[33] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks," *Med. Image Anal.*, vol. 54, pp. 30–44, May 2019.

[34] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," 2016, *arXiv:1605.09782*.

[35] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 733–742.

[36] J. Ryan Medel and A. Savakis, "Anomaly detection in video using predictive convolutional long short-term memory networks," 2016, *arXiv:1612.00390*.

[37] S. Li, K. Li, and Y. Fu, "Temporal subspace clustering for human motion segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4453–4461.

[38] R. Vidal, "Subspace clustering," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 52–68, Mar. 2011.

[39] M. Jain and S. K. Singh, "A survey on: Content based image retrieval systems using clustering techniques for large data sets," *Int. J. Manag. Inf. Technol.*, vol. 3, no. 4, pp. 23–39, Nov. 2011.

[40] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[41] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Mar. 2013.

[42] L. Wei, X. Wang, A. Wu, R. Zhou, and C. Zhu, "Robust subspace segmentation by self-representation constrained low-rank representation," *Neural Process. Lett.*, vol. 48, no. 3, pp. 1671–1691, Dec. 2018.

[43] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[44] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[45] B. S. Murugan, M. Elhoseny, K. Shankar, and J. Uthayakumar, "Region-based scalable smart system for anomaly detection in pedestrian walkways," *Comput. Electr. Eng.*, vol. 75, pp. 146–160, May 2019.

[46] Y. Yao, X. Wang, M. Xu, Z. Pu, E. Atkins, and D. Crandall, "When, where, and what? A new dataset for anomaly detection in driving videos," 2020, *arXiv:2004.03044*.

[47] Y. Yao, M. Xu, Y. Wang, D. J. Crandall, and E. M. Atkins, "Unsupervised traffic accident detection in first-person videos," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 273–280.

[48] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[49] C. Zhang *et al.*, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. AAAI. Artif. Intell.*, vol. 2019, vol. 33, no. 1, pp. 1409–1416.

[50] M. Canizo, I. Triguero, A. Conde, and E. Onieva, "Multi-head CNN-RNN for multi-time series anomaly detection: An industrial case study," *Neurocomputing*, vol. 363, pp. 246–260, Oct. 2019.

Zhili Zhou (Member, IEEE) received the M.S. and Ph.D. degrees in computer applications from the School of Information Science and Engineering, Hunan University, in 2010 and 2014, respectively. He was a Post-Doctoral Fellow at the Department of Electrical and Computer Engineering, University of Windsor, Canada. He is currently a Professor with the School of Computer and Software, Nanjing University of Information Science and Technology, China. His current research interests include multimedia security, artificial intelligence security, information hiding, blockchain, and secret sharing.

Xiaohua Dong received the B.S. degree from the Jiangsu University of Science and Technology, China, in 2019. She is currently pursuing the M.S. degree with the Nanjing University of Information Science and Technology, China. Her research interests include video processing and information security.

Zhetao Li (Member, IEEE) received the M.Eng. degree in pattern recognition and intelligent systems from Beihang University in 2005, and the Ph.D. degree in computer application technology from Hunan University in 2010. He is a Professor with the College of Information Science and Technology, Jinan University.

Keping Yu (Member, IEEE) received the M.E. and Ph.D. degrees in global information and telecommunication studies from the Graduate School of Global Information and Telecommunication Studies, Waseda University, Tokyo, Japan, in 2012 and 2016, respectively. He is currently a Researcher with the Global Information and Telecommunication Institute, Waseda University.

Chun Ding received the graduate degree. He is currently pursuing the M.S. degree with the Nanjing University of Information Science and Technology, China. His research interests include video processing and information security.

Yimin Yang (Senior Member, IEEE) received the Ph.D. degree from the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2013. He is currently an Assistant Professor with the Department of Computer Science, Lakehead University, Canada. His current research interests include artificial neural networks, signal processing, and robotics.