Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **10**, Number **3**, August **2025**

Video Anomaly Detection and Early Warning Model Based on Multi-source Data and Deep Learning

Zhen-Yu Huang[1], Zheng-Guo Yuan[2,*], Yu-Xin Duan[1], Li Ma[1], Xin-Cheng Wan[1], Yi-Chen Wang[3]

[1]Jiangxi Provincial Emergency Early Warning Center, Nanchang 330000, P. R. China
hzhenyu2024@163.com, 852772597@qq.com, 394761525@qq.com, 562410339@qq.com

[2]Jiangxi Meteorological Data Center, Nanchang 330000, P. R. China
422577658@qq.com

[3]Columbia University, 116th St & Broadway, New York 10027, USA
w.yichen@columbia.edu

*Corresponding author: Zheng-Guo Yuan
Received May 13, 2024, revised September 19, 2024, accepted December 25, 2024.

Abstract. Aiming at the problem that the current neural network models used for video anomaly detection have poor storage capacity, which leads to the easy loss of feature information and reduces the detection accuracy, the research on neural network storage capacity is carried out, and the spatio-temporal memory guided dynamic balanced edge learning video anomaly detection network (SMDNet) is designed. Based on ConvLSTM, the spatio-temporal memory module (ConvLSTM-SMM) is proposed to compensate for the problem of insufficient neural network storage capacity and applied to video anomaly detection based on deep learning framework. The designed spatio-temporal memory module continuously updates the feature parameters in the memory network by querying the features, and matches the updated feature parameters with the query features to store them, so as to obtain new feature parameters to enhance the spatio-temporal features of the network. At the same time, in order to ensure the stability of the training network, based on edge learning, the Dynamic Balanced-edge-learning Model (DBM) is proposed, which uses the spatio-temporal memory module to guide the dynamic balanced edge learning, and adopts different sample spacing in different training phases of the samples to help the network to get the appropriate edge distance in different training periods, so that the detection accuracy can be improved. For the traditional video anomaly detection methods rely on a single data source. We will combine multiple sources of data, including video images, sensor data, etc. The experimental validation part is carried out on the most challenging public datasets (Avenue, ShanghaiTech), and the experimental results show that the designed spatio-temporal memory-guided edge learning video anomaly detection algorithm has improved the video anomaly detection rate and superior detection performance.
Keywords: Video anomaly detection; Temporal memory; Dynamically balanced edge learning; LSTM

1. **Introduction.** With the development of the information age, human society has brought about some drawbacks while moving towards progress. The current social form is gradually showing a trend of polarisation, although social conflicts have eased compared with those of a few decades ago. However, due to the advancement of the current era, there are more and more kinds of social risks and new conflicts are becoming more and more complicated. It is obviously difficult to solve the new problems if the old methods

are still used. For the safety of people's health, property, etc., people are slowly starting to use electronic surveillance device. However, today's traditional cameras do not meet the current needs of society. For this series of problems, designing a video anomaly monitoring system with strong detection capability is the best solution. This system can obtain target information through real-time monitoring, automatically detect abnormalities appearing in the target and send out alarms. If the system can have a substantial increase in detection accuracy relative to traditional detection methods, it can automatically eliminate the problems of manpower verification as well as multiple cameras. This will not only improve the completion of surveillance, but also save many human and material resources. However, even if the above points are achieved, there are still serious challenges to be faced. The first one is the determination of abnormal behaviours. In different scenarios, abnormalities are defined differently and therefore need to be redefined in different environments. Secondly, in many cases, anomalies and normality may be just a fraction of a second away, and anomalies may occur in some few frames in the same video, which is very difficult to detect. Finally, anomaly data is very difficult to collect. When applying neural networks, the more anomaly data is available the better for video anomaly detection research. However, anomalies are not common in daily life, so it is difficult to collect such data. Therefore, there is still a need for more scholars to invest in the research direction of video anomaly detection in order to promote its development and better application in the current society.

1.1. **Related work.** Anomaly detection in video has been studied by many scholars as it has a wide range of applications in real-world scenarios such as intelligent surveillance, violence alerts, and evidence investigations, etc. In the 1960s, Grubbs [1] defined an anomaly as "a significant departure from the rest of the sample in the same category". Since anomalous events are rare in common environments, anomalies are often defined as patterns of behaviour or appearance that differ from common patterns in previous work [2, 3]. In anomaly detection, most of the studies have addressed the problem under the assumption that anomalies are rare or invisible and also behaviour that deviates from the normal pattern is considered as anomalous. They attempted to encode the regular patterns and identify anomalies as outliers through various statistical models. In early studies on the detection of abnormal behaviour in crowds, manual feature descriptors [4, 5] were often used to represent pedestrian appearance and motion features, and low-level visual features, such as gradients [6] and optical flow [7], were acquired by specific methods. With the significant improvement of computer performance and the successful application of deep learning in computer vision [8], many scholars have begun to try to apply deep learning to the field of video anomaly detection. In traditional video anomaly detection with handmade features, discriminative handmade feature extraction is required. Adam et al. [9] used exponential distributions to describe regular local histograms of optical flow. Deep learning approaches have proven their success in many computer vision tasks as well as anomaly detection. These include deep selfencoder based [10], Recurrent Neural Network (RNN), Convolution Neural Network (CNN), etc. In the last decade, deep learning has made significant progress in anomaly detection, allowing spatial and temporal features of objects to be learnt by performing convolutional operations on video frames and detecting various anomalies through temporal and spatial correlations. In recent years, Generative Adversarial Network (GAN) can generate data that is similar to the original data, which can be referenced for anomaly detection tasks. And more and more researchers apply GAN to video anomaly detection. Chen et al. [11] proposed an end-to-end pipeline called NM-GAN which assembles an encode-decoder reconstruction network and a CNN-based discrimination network in a GAN-like architecture. Deep

Neural Networks learn spatial features well, while Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) are commonly used for modelling sequence data. Some scholars proposed Convolutional Long Short-Term Memory (ConvLSTM), whose core is the same as LSTM, taking the output of the previous layer as the input of the next layer. The core difference is that the multiplication is replaced by convolution. It not only has the characteristics of LSTM in acquiring temporal features, but also extracts spatial features through convolution, thus acquiring spatio-temporal features. How to reduce the demand of network models on computational resources in deep learning is an important issue. Currently, research is mainly conducted in hardware and software. The following designs are often adopted: (1) Computational offloading [12], i.e., the model is placed in different parts of the edge nodes and the cloud according to the complexity, and the model is jointly optimised; (2) Specialised design [13], i.e., the use of field-programmable gate array (FPGA) to accelerate sparse matrices; (3) Dynamic model selection [14], which improves resource utilisation and reduces device power consumption by generating multiple different convolutional kernels for each layer of the network, and dynamically selecting the appropriate number of convolutional kernels based on device requirements and user constraints. However, there are many problems in hardware design. The following methods are generally used: (1) Compact model [15]: by selecting a reliable compact model, the model framework is avoided to be bloated while satisfying the accuracy. (2) Tensor decomposition [16]: neural networks have many redundant parameters, through matrix decomposition and other methods to reduce the redundancy of the network, reduce the size of the network, from where to accelerate the network model inference. (3) Knowledge distillation [17]: improves the performance of the student model by transferring knowledge from a larger, more complex model (called the "teacher model") to a smaller, simpler model (called the "student model") so that the smaller model can maintain a certain level of performance while having a smaller model. (4) Model quantisation [18]: compresses the original network by reducing the number of bits in the weights, with the disadvantage of losing some network accuracy. (5) Network thinning [19]: includes weights and neuron pruning, the former reduces the association between neurons and the latter reduces the neurons. These methods are less expensive and easier to deploy into devices compared to hardware design, which can make the model run faster on the same hardware device, thus improving the performance and efficiency of the model.

1.2. **Motivation and contribution.** Although video anomaly detection algorithms based on deep learning have made great progress, there are still some shortcomings, for example, how to effectively save video information, especially saving key information in the video is one of the keys to the detection task. The video anomaly detection task requires the network model to have strong storage capacity, while the general neural network model has poor storage capacity and feature information is easily lost. In addition, the features are often very similar within the network, and the features of the same type of video frames are very similar. Commonly used edge learning methods use fixed edge constraints as well as distance formulas, which can lead to poor sample separation. Since the final result of anomaly detection is binary in nature, the result can only be either anomalous or normal. Whether it is the lack of network information storage capacity, the inability to correctly distinguish normal or abnormal in similar videos, or the poor edge separation effect, this will lead to errors in the final detection results. This paper discusses the deep learning-based video anomaly detection algorithm feature storage and separation ability to explore and research, by improving the network model storage and sample separation ability to improve the detection accuracy of the network model by analysing the existing model storage capacity and sample separation ability of the shortcomings of this paper,

this paper proposes spatio-temporal memory-guided dynamic balanced edge detection algorithm. Combining the ConvLSTM with the memory module enhances the information reliability, and at the same time, through the dynamically balanced edge learning module DBM, the dynamic balance of the distance between different samples is maintained in the sample space, which achieves the purpose of improving the detection accuracy of the network model.

2. **Analysis of relevant principles.**

2.1. **Video anomaly detection descriptions.** Anomalies in a scene usually refer to the fact that certain events do not match the normal situation when they occur in a particular environment [20]. Since anomalous events are usually rarer than normal events, the number of negative examples (anomalous events) is usually smaller than the number of positive examples (normal events) in anomaly detection. In addition, anomalous events usually contain many different types of anomalies, which all increase the difficulty of detecting anomalous events. Another challenge is that the categories of human activities are well-defined, e.g., walking is walking and running is running. However, the concept of localisation of anomalous events is more ambiguous because anomalous events involve a wide variety of activities. In addition, the impact of fixed and dynamic cameras on anomaly detection can be very different. Due to the over-complexity of dynamic studies, in this paper, we only study video anomaly detection with static cameras. Video anomaly detection and localization depend on the complexity of the environment and the type of anomaly. The environment can be divided into the following types: sparse congestion (10 square feet per person), moderate congestion (4.5 square feet per person), and dense congestion (2.5 square feet per person) [21]. Video anomalies can be divided into single entity anomalies (such as wandering, intrusion), dual entity anomalies (such as fighting), and group anomalies (such as crowd dispersion caused by stampedes or explosions). Different types of anomalies have varying temporal and spatial complexity. Local abnormal activity refers to activities that significantly deviate from their neighboring activities in time and space. Global abnormal activities refer to activities that occur in an abnormal manner on a global scale [22]. Inter-point anomalies are usually defined as a distribution of a data point in a data set that is significantly different from other data points and can be considered as a random irregularity. Some data points deviate significantly from others, resulting in context-specific anomalies.

2.2. **Video anomaly detection methods.** Semi-supervised methods combine the advantages of both supervised and unsupervised methods, using both labelled and unlabelled data for training. Active learning, on the other hand, is an approach that progressively reduces the amount of labelled data required, maximising the use of the labelled budget by querying and labelling the unlabelled data. Generative features of GANs with active learning capabilities have been successfully applied for outlier detection [23].

2.3. **Network infrastructure.** Convolutional layer is one of the most important components of many neural networks, which extracts features from the input data by using a set of learnable convolutional kernels. Convolutional layer is one of the core components in neural networks and is mainly used to extract features from the input data. As Equation (1) is the formula usually used for convolution:

$$y = f(W * x + b) \tag{1}$$

where x denotes the input data, $*$ denotes the convolution operation, W denotes the convolution kernel, b denotes the bias vector, and y denotes the output feature map.

The commonly used neuron activation functions are Tanh as in Equation (2) and Sigmoid function as in Equation (3), which are saturated nonlinear functions, which makes the convergence rate slow when training the network using gradient descent algorithm. In contrast, using a non-saturated function as an activation function, such as the rectified linear unit (Relu) in Equation (4), can significantly speed up training. Relu is a simple nonlinear function that ensures that the output is always positive by thresholding the input. In the model of this paper relu is used as the unique activation function for all layers except the output layer. Where x in the format all denotes the input and the left side of the equation denotes the output.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2}$$

$$S(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

$$f(x) = \max(0, x) \tag{4}$$

The output layer is different from the other layers in that its output is presented as a probability sum of 1, which indicates the confidence level of the selected category, with a higher probability value indicating a higher level of confidence. Commonly used functions are softmax, tanh function.

$$\text{softmax}(z_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{5}$$

where z_j denotes the jth element of the input vector, K denotes the length of the vector z, and the softmax function converts each element in the vector z to a probability value such that the probability values of all the elements sum to 1.

Inverse convolution is a common operation used to convert the output of a generator into an image. Inverse convolution is an inverse operation that converts a low-dimensional feature mapping into a high-dimensional feature mapping, this operation is also known as transpose convolution or inverse convolution layer.

The inverse convolution layer is usually used in Generator, the inverse convolution layer plays a vital role in GAN, it can convert the low-dimensional feature maps into high-dimensional feature maps.

$$x = f(W^T * y + b) \tag{6}$$

where W^T denotes the anti-convolution kernel, $*$ denotes the convolution operation, $f(y)$ denotes the output of the previous layer, b denotes the bias term, and x denotes the result of the anti-convolution.

2.4. **ConvLSTM.** Each LSTM unit in ConvLSTM consists of three gates (input gate, output gate, and forgetting gate) and a memory unit for controlling the flow of information and preserving the previous state. In addition, ConvLSTM includes a number of convolutional layers that are used to perform convolutional operations on the input data to extract spatio-temporal features.

The core essence of ConvLSTM is the same as LSTM, which takes the output of the previous layer as the input of the next layer. The overall operation is shown in Equations (7) to (11).

$$I_t = \sigma(W_i * X_t + U_i * H_{t-1} + b_i) \tag{7}$$

$$F_t = \sigma(W_f * X_t + U_f * H_{t-1} + b_f) \tag{8}$$

$$O_t = \sigma(W_o * X_t + U_o * H_{t-1} + b_o) \tag{9}$$

$$C_t = F_t \odot C_{t-1} + I_t \odot \tanh(W_c * X_t + U_c * H_{t-1} + b_c) \tag{10}$$

$$H_t = O_t \odot \tanh(C_t) \tag{11}$$

Symbol conventions: X_t is the current input feature map, W_i is the convolution kernel, I_t is the output of the input gate, H_{t-1} denotes the hidden state of the previous time, b_i denotes the bias term, σ denotes the sigmoid function, $*$ denotes the convolution operation, F_t denotes the output of the oblivion gate, O_t denotes the output of the output gate, W_f, U_f, b_f, W_o, U_o, b_o denote the convolution kernel and bias term of the oblivion gate and output gate respectively. C_t denotes the cell state at the current time step, $\odot$ denotes element-by-element multiplication, W_c, U_c, b_c are the convolution kernel and bias term used to update the cell state, and H_t denotes the output at the current time step.

3. Spatiotemporal memory guided dynamic balanced edge learning for video anomaly detection.

3.1. Temporal memory module.

LSTM can obtain excellent temporal information. The details are as follows:

The deep spatio-temporal features obtained by ConvLSTM are used as input information for querying features in the spatio-temporal memory module SMM, which are divided into several low-dimensional feature vectors, denoted as Q_ℓ^i ($i \in [1, K]$). Q_ℓ^i is read by different image channels C (different pixels correspond to different image channels) of size $1 * 1 * C$. The obtained Q_n^i is fed into the SMM, and new Q_n^i data are stored while updating the memory features P_m in the SMM. ($m = 1, 2, ..., 10$). Initially the stored data P_m in the ConvLSTM-SMM is replaced with a random value, and the memory P_m is updated with different Q_ℓ^i after starting the training. (where $K = H * W$, H is the height of the image, W is the width of the image, C is the number of image channels, and t denotes the number of frames corresponding to the video frames.)

(1) Reading feature information

Figure 1 shows the process of ConvLSTM-SMM reading and storing feature information. After multiplying the feature vector Q_ℓ^i output by the query feature module with the feature information P_m ($m = 1, 2, ..., 10$) stored in the memory module, the similarity between the feature vectors is calculated to obtain the matching probability $r_{t,m}^k$ between Q_ℓ^k and the corresponding P_m. The calculation process is shown in Equation (12), where the total matching probability r_t is 1. The feature vector after reading in new feature information is denoted as $\hat{P}_t^k$, and its calculation process is shown in Equation (13), completing the reading of feature information.

$$r_t^{k,m} = \frac{(P_m^k)^T Q_\ell^k}{\sum_{n=1}^{M=10} (P_n^k)^T Q_\ell^k} \tag{12}$$

$$\hat{P}_t^k = \sum_n^M r_t^{k,n} p_n^k \tag{13}$$

(2) Updating feature information

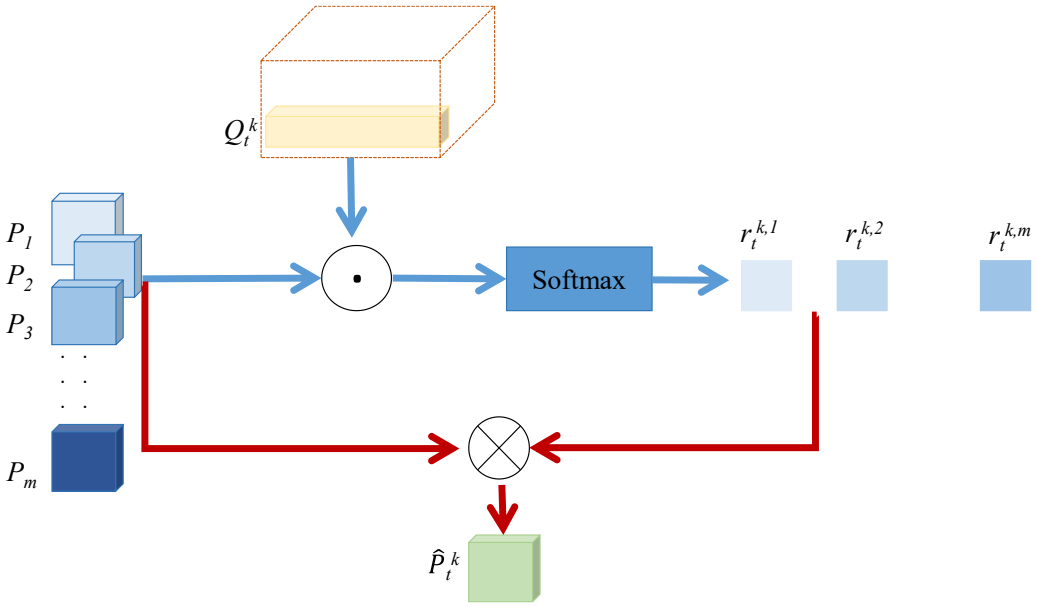

Figure 1. Feature information reading into memory

The updating of feature information is similar to the reading of feature information as described above, but the difference lies in the acquisition of matching probability. Each P_m is assigned multiple Q_ℓ^i, and we need to obtain the corresponding index set index for each P_m, and the matching probability in each index set is denoted as V_t. The current index set is normalised to the maximum value, and the calculation process is shown in Equation (14). Finally, the new updated data is obtained by softmax.

In this paper, the ratio of the update parameter P (memory) to the query reading parameter Q (query) is chosen as $P : Q = 10 : 2000$. For the mth item of memory information, the mth item of memory information can be updated by finding the corresponding query index of the mth item.

Since more than one Q_ℓ^i can be assigned to the same memory information P_m, there may be more than one query to update a single memory information. The update is performed as in Equation (15), where v' is obtained from Equation (16). (D_t^m denotes the index set of the mth query feature in P).

$$V_t^{k,m} = \frac{(P_t^m)^T Q_\ell^k}{\sum_{n=1}^{N=2000} (P_t^m)^T Q_\ell^n} \tag{14}$$

$$v_t^{k,m} = \frac{V_t^{k,m}}{\max\limits_{k' \in D_t^m} V_t^{k',m}} \tag{15}$$

$$P^m = L2_{norm}\left(P^m + \sum_{i \in D_t^m} V_t^{i,m} Q_\ell^i\right) \tag{16}$$

In the data preprocessing stage the samples are classified as normal samples, abnormal samples, so the data processed in the SMM network are either all normal samples or all abnormal samples. The acquired feature vector $\hat{P}_t^k$ is concatenated with Q_t along the dimensions of channel C to obtain new feature information, which is later fed to the decoder and the DBM.

3.2. **Dynamic equilibrium edge learning.** The DBM proposed in this paper divides all the samples into 3 parts, anchor and positive are taken from the normal samples and negative is taken from the abnormal samples, after the features are obtained by the coding network, the features are fed into the DBM. Edge Learning uses the constraints to increase the distance between the normal samples and the abnormal samples, so that the abnormality detection is easier to differentiate the abnormal samples.

Since there are only two kinds of samples in the anomaly detection study, normal and abnormal samples, the dynamic balanced edge learning studied in this paper uses a ternary loss function for parameter calculation. As mentioned earlier, training with the ternary loss function makes the same kind of samples (positive) closer to the anchor and different kinds (negative) further away from the anchor, where the anchor is a randomly selected sample from the training data. In other words, the ternary loss function can minimise the distance between the anchor and the same type of samples, and maximise the distance between the anchor and different types of samples, so as to better distinguish between normal and abnormal samples. The distance between samples is calculated using Euclidean distance.

Excessively large values of edge constraints make the model training more difficult, which in turn results in non-convergence of the network. So, in the initial stage of training, the training data sample distribution is relatively uniform, and the distance between the samples is relatively close, using a fixed and smaller edge of the method to open the distance between the different categories of samples; when most of the samples to distinguish the distance, the traditional edge of the fixed constant method fails, you need to use new edge constraints method, increase the maximum edge constraints. In terms of edge constraints this paper proposes to use the form of segmentation function to set the edge constraints to replace the original fixed edge conditions. For the edge constraints are set as shown as follow.

$$\text{margin loss} = \max(0, \|f^a - f^p\|_2^2 - \|f^a - f^n\|_2^2 + \psi) \tag{17}$$

$$a = |f^a - f^n|,\ b = |f^a - f^p| \tag{18}$$

$$\psi = \begin{cases} a - b & (1 < a - b \leq 1.5) \\ 1.5 & (1.5 < a - b) \\ 1 & (a - b \leq 1) \end{cases} \tag{19}$$

Different formulas can be used to achieve different training purposes between samples. Even if anchor, positive, negative is selected twice consecutively, the dynamic edge constraint can continue to train the samples and increase the sample interval.

The edge constraint coefficient ψ is between 1 and 1.5. When the maximum value is around 1.5, it almost reaches its threshold, and the overall framework efficiency can hardly be improved.

Therefore, such an edge constraint will form a dynamic balance, which has the advantage of not letting the samples always be fixed at the same edge, and constantly updating the edge distance to help different samples to find the most suitable distance, the original fixed edge constraint will make the training efficiency decline when the training reaches a certain number of rounds because of the instability of training. The dynamic balanced edge constraint solves this problem to some extent by improving the stability of training.

3.3. **Loss function.** The overall loss function can be expressed as in Equation (20):

$$L = L_{pred} + \lambda L_{margin} + \gamma L_{separateness} \tag{20}$$

where λ and γ parameters are used 1, λ and γ are the percentage of balancing its loss, and the values can be modified in different environments. L_{pred} is predict loss, L_{margin}

is margin loss, and $L_{separateness}$ is separateness loss. The descriptions are given next, respectively.

(1) Predict loss

Symbol convention: I_q^a denotes a predefined random anchor sample anchor, and normal frames are selected as the anchor in the experiment; I_l^p normal frame samples, I_l^n anomalous frame samples, $\hat{}$ denotes the output obtained from prediction.

In the prediction network, cross-loss is used to measure the difference between the predicted value and the real value. In the experiment, we follow the principle of "supporting normal samples and distorting abnormal samples": if both I^p and I^a are normal samples, adding I^p to I^n (i.e., closer to normal samples), which is an abnormal sample with a larger error, may distort the normal samples and increase the prediction error. The prediction loss is calculated as in Equation (21):

$$L_{pred}(\hat{I}_{l+T}^a, \hat{I}_{l+T}^n) = \|\hat{I}_{l+T}^a - I_{l+T}^a\| + S_{l+T}\|\hat{I}_{l+T}^n - I_{l+T}^n\| \tag{21}$$

The computation of prediction error is based on video clips, i.e., T frames are taken consecutively from the test video to form a video clip. In the experiment, we set $T = 4$. In Equation (21), S_t denotes the degree of abnormality $\{S_t \in [0, 1]\}$.

If there are more than two abnormal frames in a T-frame video segment it is defined as an abnormal video segment and vice versa. In the experiment, in order to make the prediction value close to the real value, I^p is not taken into account in the prediction error calculation and $S_t = 1$ is set.

(2) Separateness loss

Separateness loss is applied to the Spatio-temporal Memory Module (SMM) reading and updating process, which serves to assign similar query features to the corresponding memory information. The SMM will constantly and repeatedly update the information in the memory when extracting the features, which makes the similarity of the neighbouring memory information higher.

However, the anomaly detection task requires each item in the memory to be as far away from each other as possible to reduce the error when dealing with various forms of normal data, so the feature separation loss is invoked in the paper, and the formula for the feature separation loss function is shown in Equation (22).

$$L_{separateness} = \sum_{t=1}^{T}\sum_{k=1}^{K}\left(\left\|Q_t^k - P_{x_1}\right\|_2 - \left\|Q_t^k - P_{x_2}\right\|_2 + a\right) \tag{22}$$

where T denotes the length of the video frame, $K = H * W$ (multiplication of the number of columns and rows of the frame), a is a fixed edge, and P_{x_1}, P_{x_2} denote the closest memory information and the second closest memory information to the query vector in terms of similarity, respectively. In order to distinguish the query vector from the closest memory information in the storage space from other memory information.

Separation loss encourages the query vector to be close to the closest memory information and at the same time away from the second closest memory information by means of triple loss to help update the best memory information in the memory. In turn, it can solve the similarity problem between different items in memory and enhance the information discrimination in memory.

(3) Margin loss

As mentioned earlier, the dynamic edge module gives the calculation of the edge loss, so the edge loss L_{margin} can be expressed as follow:

$$L_{margin} = \max\left(0, \|f^a - f^p\|_2^2 - \|f^a - f^n\|_2^2 + \psi\right) \tag{23}$$

3.4. **SMDNet framework.** By combining the features of DBM and ConvLSTM-SMM, the memory items are updated in SMM, the spatio-temporal features are extracted alternately and then combined with DBM to form SMDNet, and the normal frames and abnormal frames are divided into different clusters and sent to the network for training, so as to improve the performance of network detection, shown as Figure 2.

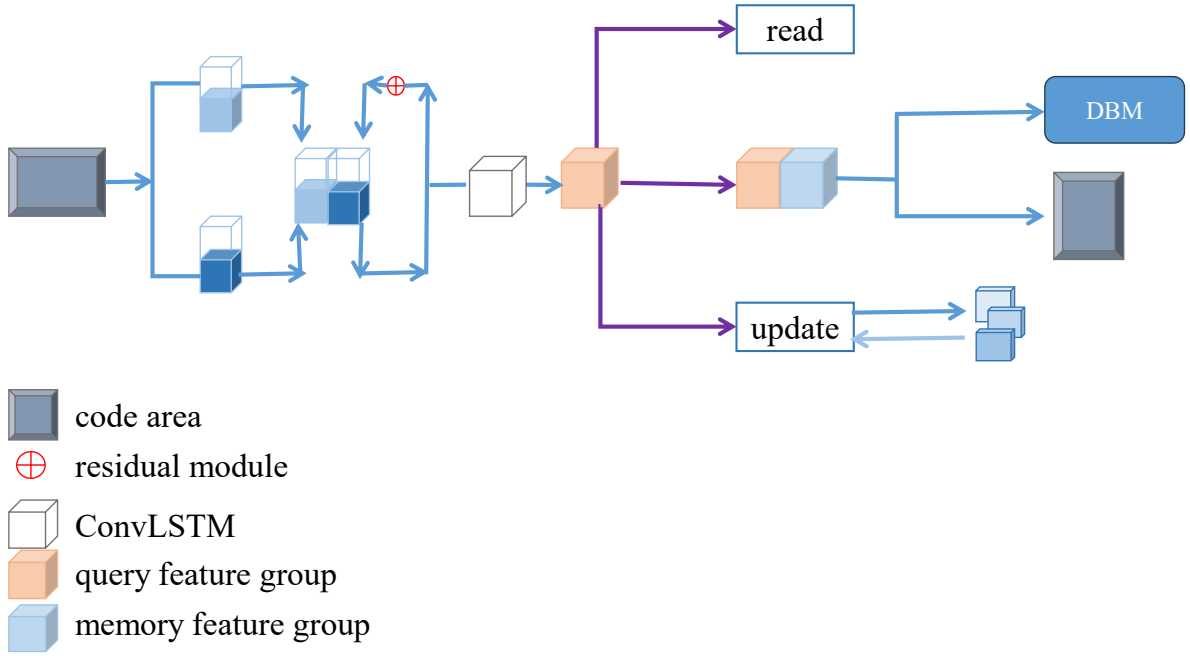

Figure 2. SMDNet overall framework

4. Experimental design and analysis of results.

Table 1. Server information

Accessory Name	Model
Operating platform	Windows Server 2016
CPU	Intel Xeon Bronze 3106
RAM	64G
GPU	NVIDIA GeForce GTX 1080TI 11GB
Deep learning framework	Tensorflow 1.14
Learning rate	[0.0002, 0.00002]
Gradient descent method	Adam

4.1. **Experimental environment.** All the algorithms in this paper use python code and tensorflow deep learning framework. And all the experiments in this paper are done in the same server environment, the server environment and configuration are shown in Table 1.

4.2. **Data processing.** In this paper, video frame-level annotations and small batch samples are used to mark the frame-level samples (frame-level). Firstly, the video is edited into frames, and the adjacent video frames are selected to form a video segment, usually a video segment length is 4. Due to the different sizes of different video frames, the experiments need to resize all the video frames to [224, 224, 3], and the image channel

is adjusted to RGB. According to the number of abnormal frames, the training set data is divided into normal video segments and abnormal video segments. Divide the abnormal data in the test set into K folds, each fold contains only some abnormal events, not all of them. Usually, $K = 10$ or 5. In this paper, we choose to put the first fold into the training set and the rest as the test set, which ensures that the test set must contain the abnormal events that are not included in the training set, and the test set may also contain the types of anomalous events that are observed in the training set.

4.3. **Evaluation indicators.** The evaluation index follows the commonly used index for anomaly detection, AUC, i.e., the area under the ROC of the subject's work characteristic curve. A larger AUC area means a better performance. In this paper, we calculate the AUC by calculating the anomaly score curve.

After the model is trained, the image judgement score Score is obtained by predicting the quality of future frames, then the degree of abnormality is measured according to the score, and finally the frame-level AUC is calculated according to the results of image frame prediction and GT. The quality of video frames is measured by Peak Signal to Noise Ratio (PSNR), and the judgement score Score is calculated. The calculation formula is shown as follows:

$$P_t(I_t, \hat{I}_t) = 10 \log_{10} \frac{1}{\frac{1}{N} \sum_{i=0}^{N} \sum_{j=0}^{M} \|I_t(i,j) - \hat{I}_t(i,j)\|_2^2} \tag{24}$$

$$S_x = \frac{P_t(I_t, \hat{I}_t)}{X}, \quad S_y = \frac{P_t(I_t, \hat{I}_t)}{Y} \tag{25}$$

In Equation (24), I_t is the current frame; i, j are the rows and columns of the pixel position of the t-frame; N and M denote the number of rows and columns; and P denotes the peak signal-to-noise ratio. In Equation (25), S denotes Score; X is the number of normal samples in the dataset ($GT = 0$), and Y is the number of abnormal samples in the dataset ($GT = 1$), and S_x, S_y are the corresponding scores in the corresponding dataset.

4.4. **Ablation experiment.** In order to verify the effectiveness of each module in the network, this chapter carries out ablation experiments on the Avenue dataset, by demonstrating the performance of Score as well as AUC for normal and abnormal samples. According to the quantitative comparison of each module of the model, when the difference of Score Gap is bigger, the differentiation of the samples is higher, except for the abnormal score, which is the lower the better, the higher the better. Table 2 shows the impact of different module units on the overall framework. Bold indicates the best performance.

From Table 2, taking the Avenue dataset as an example, it can be seen that the adopted modules have unequally improved the performance of the network framework to a certain extent, and an AUC of 94.1% is obtained when only ConvLSTM is available, which proves the importance of ConvLSTM for spatio-temporal feature extraction, and when DBM is added based on this, the dynamically balanced edges, which characterise the normal-normal samples and the normal-abnormal sample interval between normal-normal samples and normal-abnormal is expanded again, so the AUC grows to 94.8% again. When DBM is replaced by SMM, due to the advantage that SMM can store informative features, it can be seen that compared to the previous two, the performance reaches 95.1%, and finally all the modules are united, so that the final efficiency reaches 95.9% of the AUC, which is still an advantage of this method in the ShanghaiTech dataset.

Figures 3 and 4 show the AUCs obtained for the Avenue and Shanghaitech datasets at different rounds, respectively. in the case of Avenue, for example, the method in this paper can achieve a more stable AUC of about 95% in a limited number of training rounds

Table 2. The impact of different modules on the overall network framework

Datasets	ConvLS	SMM TM	Memory	DBM	Normal frames Score	Abnormal frames Score	Score gap	AUC
Avenue	×	×	×	0.774	0.467	0.312	87.0%	
Avenue	✓	×	×	0.811	0.290	0.524	94.1%	
Avenue	✓	×	✓	0.814	0.262	0.555	94.8%	
Avenue	✓	✓	×	0.799	0.201	0.596	95.1%	
Avenue	✓	✓	✓	**0.833**	**0.188**	**0.642**	**95.9%**	
Shanghai	×	×	×	0.634	0.473	0.181	77.0%	
Shanghai	✓	×	×	0.738	0.569	0.171	77.3%	
Shanghai	✓	×	✓	0.725	0.543	0.183	77.6%	
Shanghai	✓	✓	×	0.731	0.554	0.179	77.4%	
Shanghai	✓	✓	✓	0.726	0.533	**0.190**	**78.2%**	

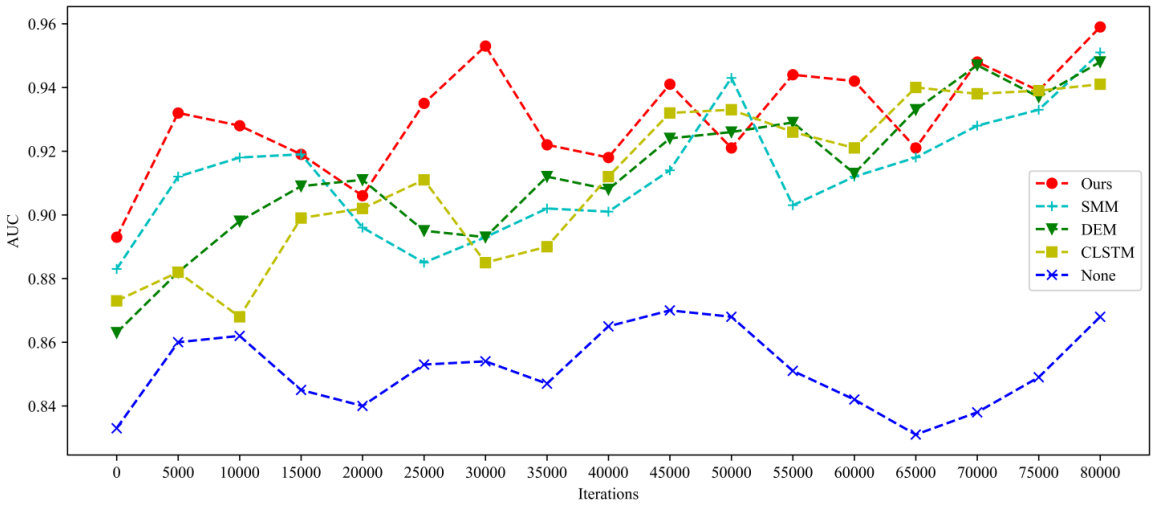

Figure 3. AUC obtained in different rounds (Avenue)

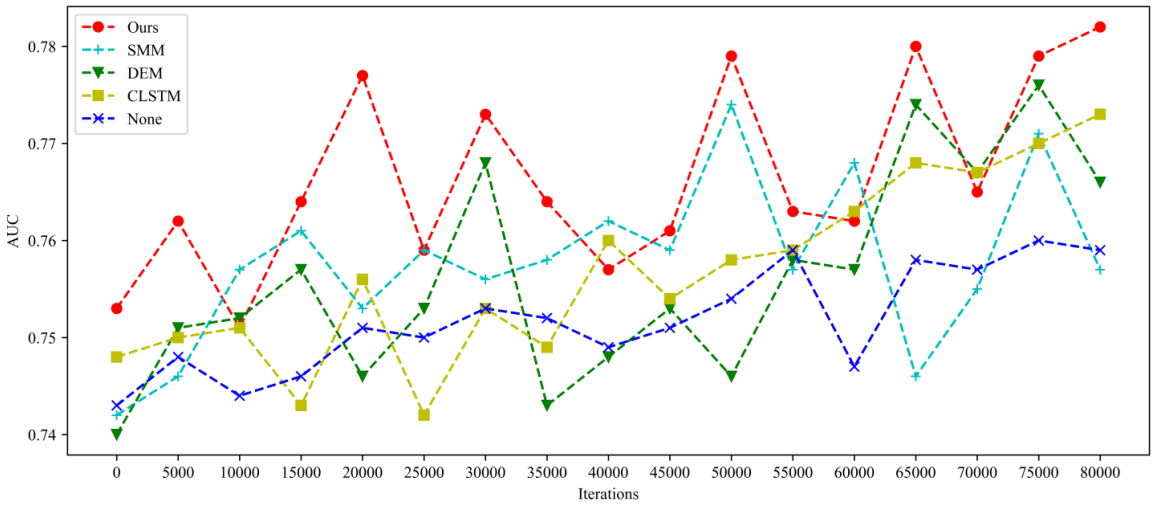

Figure 4. AUC obtained in different rounds (ShanghaiTech)

in about 20,000 rounds (about 5 hrs.), and the AUC can eventually reach about 95.9% by zigzagging through the continuous training. AUC.

As a result, it can also be seen from the data in the above figure that the innovation of this paper is in line with it, and the SMM gets better spatio-temporal features so as to improve the overall evaluation efficiency. Moreover, according to the characteristics of DBM, even when it reaches a certain AUC value, due to its dynamic edge characteristics, it still tries to find a better weight ratio with the increase of the number of training rounds, which helps the framework to improve the AUC to a certain extent, which is in line with the argument in the previous paper.

Table 3. The impact of separation loss on performance

Separateness	Avenue		ShanghaiTech	
	Score gap	AUC	Score gap	AUC
×	0.516	95.3%	0.182	77.2%
✓	**0.642**	**95.9%**	**0.190**	**78.2%**

It can also be seen from Table 3 that the calculation of loss function by adding SMM's separation of normal and abnormal samples to calculate-separation loss is favourable for judging the score gap Score gap and AUC. Bold indicates the best performance.

4.5. **Analysis of experimental results.** This chapter investigates the use of ConvLSTM-SMM, DBM combined with codec network and separation loss function for future frame prediction task. By comparing the research model with existing anomaly detection models on the CUHK Avenue, ShanghaiTech dataset, the AUC, which is commonly used for anomaly detection, is used as the accuracy assessment metric.

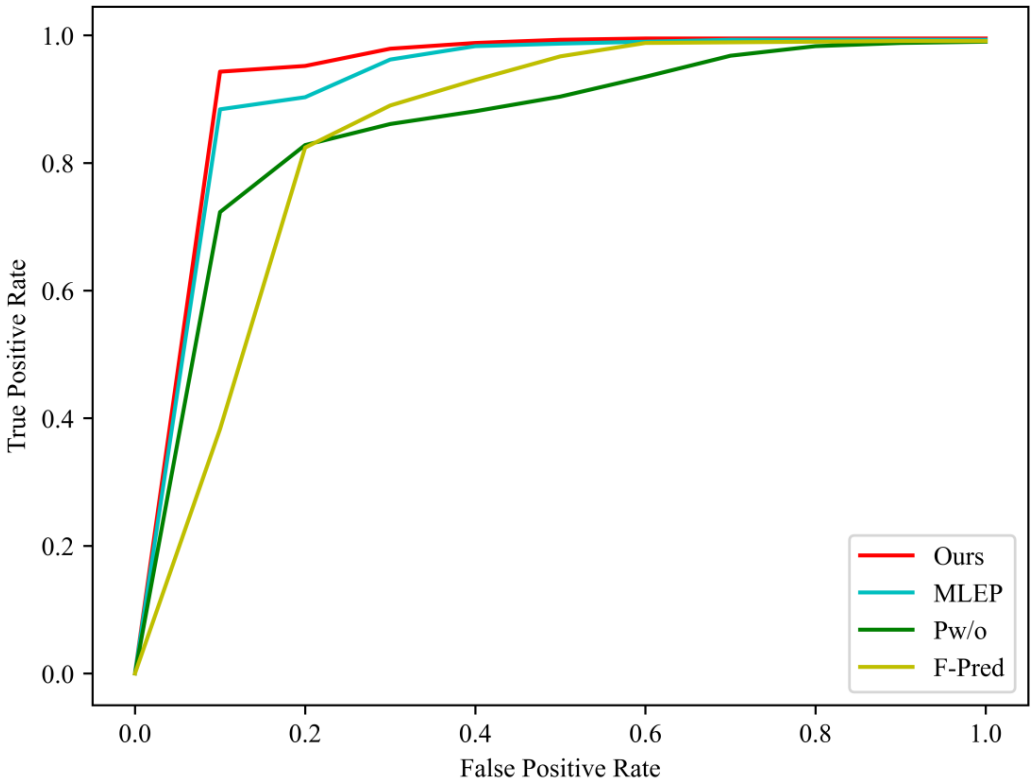

Figure 5. AUC results for the Avenue dataset test

As can be seen from Table 4, compared to other network frameworks, the proposed method in this chapter achieves a variable improvement in AUC in the Avenue and

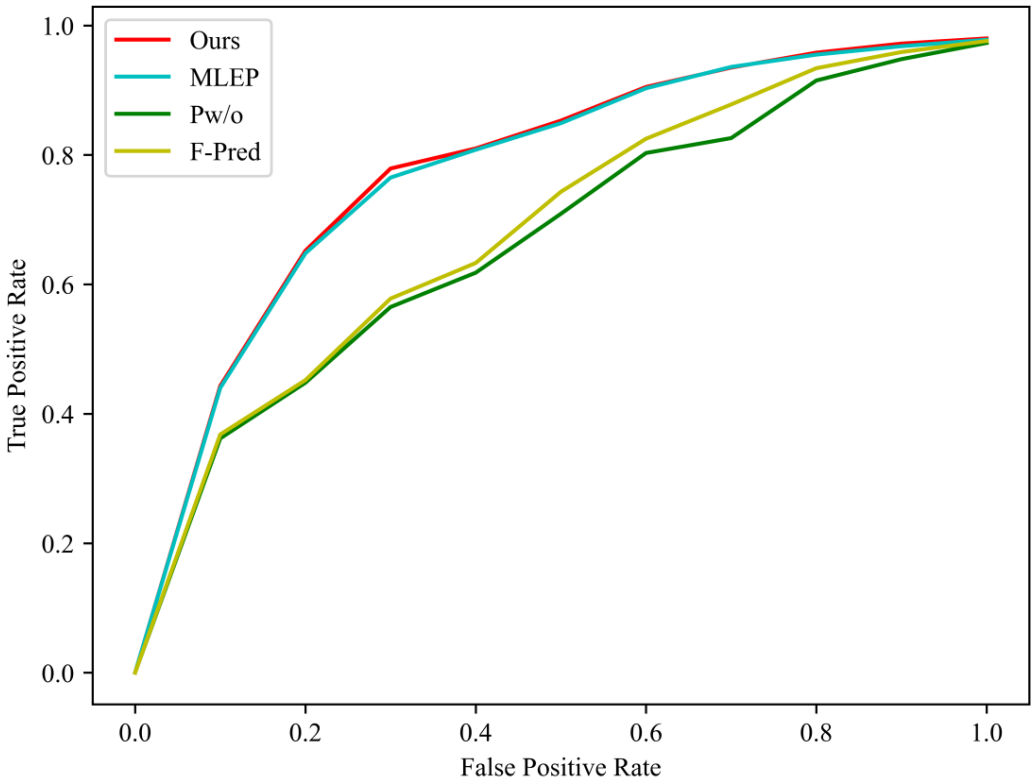

Figure 6. AUC results for the Shanghaitech dataset test

Table 4. Comparison of Abnormal Performance of Different Algorithms

Separateness	**Avenue**	**Shanghai**
Fast [24]	87.0%	-
P w/o Mem	88.5%	70.5%
MLEP	92.8%	76.8%
Ours	95.9%	78.2%

ShanghaiTech datasets, respectively. In Figures 4 and 5, the reliability of the method is also confirmed by the fact that the area under the curves is larger than that of the other methods, and bold indicates the best performance.

In real-life scenarios, the technology can be used in intelligent surveillance systems, such as among the cameras applied in schools or parks. When an abnormality is encountered, the system will promptly issue an alarm and notify security personnel to take appropriate measures to ensure the safety and order of the premises. Such an application can greatly improve the efficiency and accuracy of surveillance and provide better protection for public safety and personal property protection.

The principle is to first convert the video into video frames and learn the spatio-temporal characteristics of the video frame segments through a network model. The learned information is then used to predict the next frame and compare it with the real video frame. This method can be applied to more than a dozen video surveillance systems to detect abnormal behaviour in advance and avoid potential security risks by predicting future frames.

5. **Conclusions.** In this paper, SMDNet method for video anomaly detection is proposed. ConvLSTM method has the advantage of acquiring spatio-temporal features, but

due to the storage capacity it leads to loss of information when compression of spatio-temporal feature information is performed. In order to improve the storage capacity of the ConvLSTM method, this chapter proposes the ConvLSTM-SMM module on the basis of the ConvLSTM method, and the extracted spatio-temporal feature information is stored into the memory module. Then for the stability of training and to increase the upper limit of the detection accuracy of the network model, dynamic balanced edge learning is proposed, which makes it possible to continuously reduce the spacing of samples of the same type and increase the spacing of samples of different types on the basis of edge learning. The ablation experiment also proves the point of view of this chapter, SMM gets better spatio-temporal features, thus improving the overall evaluation efficiency, and according to the characteristics of DBM dynamic edges, it can still help the framework to improve the AUC to a certain extent by increasing the number of training rounds. Finally, the two methods are completed cascaded, which makes the two methods get good compatibility in the model. And there is a good effect improvement in both datasets. In further research, it is planned to try to improve the inference speed of the algorithm under the precondition of keeping the original detection accuracy as much as possible, in order to achieve better practicality.

REFERENCES

[1] F. E. Grubbs, "Procedures for detecting outlying observations in samples," Technometrics, vol. 11, no. 1, pp. 1-21, 1969.

[2] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," Image and Vision Computing, vol. 106, 104078, 2021.

[3] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab, "Machine learning for anomaly detection: A systematic review," IEEE Access, vol. 9, pp. 78658-78700, 2021.

[4] T. Xiang, and S. Gong, "Video behavior profiling for anomaly detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 5, pp. 893-908, 2008.

[5] E. Akagündüz, M. Aslan, A. Şengür, H. Wang, and M. C. Ince, "Silhouette orientation volumes for efficient fall detection in depth videos," IEEE Journal of Biomedical and Health Informatics, vol. 21, no. 3, pp. 756-763, 2016.

[6] D.-G. Lee, H.-I. Suk, S.-K. Park, and S.-W. Lee, "Motion influence map for unusual human activity detection and localization in crowded scenes," IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 10, pp. 1612-1623, 2015.

[7] P. Pareek, and A. Thakkar, "A survey on video-based human action recognition: recent updates, datasets, challenges, and applications," Artificial Intelligence Review, vol. 54, no. 3, pp. 2259-2322, 2021.

[8] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," Image and Vision Computing, vol. 106, 104078, 2021.

[9] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 555-560, 2008.

[10] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," Image and Vision Computing, vol. 106, 104078, 2021.

[11] D. Chen, L. Yue, X. Chang, M. Xu, and T. Jia, "NM-GAN: Noise-modulated generative adversarial network for video anomaly detection," Pattern Recognition, vol. 116, 107969, 2021.

[12] T. Jiao, "Mobile English teaching information service platform based on edge computing," Mobile Information Systems, vol. 2021, pp. 1-10, 2021.

[13] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang, "Holistic network virtualization and pervasive network intelligence for 6G," IEEE Communications Surveys & Tutorials, vol. 24, no. 1, pp. 1-30, 2021.

[14] M. de Prado, M. Rusci, A. Capotondi, R. Donze, L. Benini, and N. Pazos, "Robustifying the deployment of tinyml models for autonomous mini-vehicles," Sensors, vol. 21, no. 4, 1339, 2021.

[15] W. Sun, G. Zhang, X. Zhang, X. Zhang, and N. Ge, "Fine-grained vehicle type classification using lightweight convolutional neural network with feature optimization and joint learning strategy," Multimedia Tools and Applications, vol. 80, no. 20, pp. 30803-30816, 2021.

[16] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," IEEE Transactions on Signal Processing, vol. 65, no. 13, pp. 3551-3582, 2017.
[17] T.-Y. Wu, L. Wang, and C.-M. Chen, "Enhancing the Security: A Lightweight Authentication and Key Agreement Protocol for Smart Medical Services in the IoHT," Mathematics, vol. 11, no. 17, 3701, 2023.
[18] T.-Y. Wu, Q. Meng, L. Yang, S. Kumari, and M. Pirouz, "Amassing the Security: An Enhanced Authentication and Key Agreement Protocol for Remote Surgery in Healthcare Environment," Computer Modeling in Engineering & Sciences, vol. 134, no. 1, pp. 317-341, 2023.
[19] T.-Y. Wu, F. Kong, Q. Meng, S. Kumari, and C.-M. Chen, "Rotating behind security: an enhanced authentication protocol for IoT-enabled devices in distributed cloud computing architecture," EURASIP Journal on Wireless Communications and Networking, vol. 2023, 36, 2023.
[20] O. P. Popoola, and K. Wang, "Video-based abnormal human behavior recognition—A review," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 6, pp. 865-878, 2012.
[21] H. Jacobs, "To count a crowd," Columbia Journalism Review, vol. 6, no. 1, 37, 1967.
[22] K. Pawar, and V. Attar, "Deep learning approaches for video-based anomalous activity detection," World Wide Web, vol. 22, no. 2, pp. 571-601, 2019.
[23] Y. Liu, Z. Li, C. Zhou, Y. Jiang, J. Sun, M. Wang, and X. He, "Generative adversarial active learning for unsupervised outlier detection," IEEE Transactions on Knowledge and Data Engineering, vol. 32, no. 8, pp. 1517-1528, 2019.
[24] S. Li, Y. Cheng, Y. Liu, and Y. Yang, "Fast anomaly detection based on 3D integral images," Neural Processing Letters, vol. 54, no. 2, pp. 1465-1479, 2022.