Pattern Recognition 153 (2024) 110550

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Video anomaly detection guided by clustering learning

Shaoming Qiu, Jingfeng Ye *, Jiancheng Zhao, Lei He, Liangyu Liu, Bicong E., Xinchen Huang

Communication and Network Laboratory, Dalian University, Dalian, 116622, China

ARTICLE INFO

Keywords:
Video anomaly detection
Spatial–temporal cascade auto-encoder
Clustering learning
Memory-guided

ABSTRACT

With the fuzzy boundary between normal and abnormal video data, which cannot be well distinguished by most methods, anomaly detection in video requires better characterization of the data. First we give a convolution-enhanced self-attentive video auto-encoder based on the U-Net architecture, which can extract richer image features. Secondly we design a dual-scale feature clustering structure for this encoder, which simultaneously compresses the channel and spatial structure features of the image to represent the features to obtain good coding characteristics and expand the boundary between normal and abnormal data. We also verify that our approach is equivalent to a class of auto-encoders for memory-guided learning. Finally, in the reconstruction task, since video auto-encoders are capable of triggering temporal time leakage phenomena that can lead to network performance degradation, we propose an anomaly score computation paradigm for video auto-encoders that utilizes the average frame anomaly score of a video clip to compute the first frame anomaly score in that video clip.

Extensive experiments on three benchmark datasets show that our method outperforms most existing methods on large datasets with complex patterns. The code will be published at the following link: Anomaly-detection-guided-by-clustering-learning

1. Introduction

A video anomaly detection task (VAD) is a computer vision task that aims to detect events in a video sequence that do not conform to normal patterns or expected behavior. These events are called anomalous events and they may be due to vandalism, malicious interference, equipment malfunction, environmental changes, or other reasons [1–3]. Video anomaly detection is a challenging task in most cases: firstly, event anomalies of video can be decomposed into spatial and temporal anomalies, and some event anomalies cannot be determined by a single spatial or temporal cue, e.g., a skateboarder and a pedestrian are similar in both appearance and movement, but can be differentiated from each other by the change in posture. Secondly, in the dataset used for anomaly detection tasks, normal pattern data is easy to obtain, but anomalous pattern data is difficult to collect due to the high cost of labeling. Therefore, most of the work describes the general distribution pattern of each manual feature of image frames separately from the spatial–temporal perspective, and the image frames that significantly deviate from the general distribution pattern are regarded as anomalous frames [4–6].

In recent years, the mainstream in the field of video anomaly detection has used methods that use auto-encoders to model the spatial–temporal–temporal features of normal mode data [7–9]. These methods train the auto-encoder using only normal mode data thus putting the encoder into an over-fitting state. When the input data deviates significantly from the normal mode, the decoder outputs image frames with large reconstruction errors from the original image frames, thus enabling the detection of anomalous frames. Such methods are usually based on the U-Net architecture, and the existence of the short-cut layer greatly improves the generalization ability of the network [1,10], which is able to reconstruct the network well even in the face of anomalous pattern data, which makes the network unable to determine the boundary between anomalous data and normal data.

Some work uses only image encoders to extract features, they stack a number of picture frames in the channel direction and then perform feature extraction by a 2D convolution kernel [1,7,8,11], which improves the network's ability to encode spatial information, but makes limited use of temporal information, and works with lower performance in complex scenarios [12–14], and there is also work that uses predictive image frames forcing the encoder to learn the temporal connection between frames and achieved good results [1,15–17]. Theoretically better results can be achieved by using a video encoder, but its powerful spatial–temporal generalization capability allows the network to make sense of the video frames in real time without being trained, which instead leads to performance degradation [9]. In particular, on the reconstruction task, we found that the video encoder is capable of triggering a serious temporal event leakage phenomenon.

* Corresponding author.
E-mail address: yejingfengjack@qq.com (J. Ye).

https://doi.org/10.1016/j.patcog.2024.110550
Received 4 January 2024; Received in revised form 2 March 2024; Accepted 28 April 2024
Available online 6 May 2024

The work in this paper is based on the reconstruction/prediction task under the U-Net framework, but incorporates the Swin Transformer framework with the I3D convolution module on top of it [14,18]. In this paper, the short-cut layer of U-Net is eliminated and the Patch Embed operation in the ST framework is retained to reduce the computation, in addition, the attention mechanism is considered to be able to capture global features, but there is blurring of details in the image reconstruction task, for this reason, the I3D convolution module is introduced to assist in capturing the detailed information of the picture and improve the performance of the model. Finally we also introduce a new video anomaly detection paradigm to address the temporal event leakage problem of the video encoder on the frame-level reconstruction task, which utilizes the average frame anomaly score of a segment of video to estimate the first video frame of the segment.

Many works in the field of image classification have tried the multi-scale feature fusion of attention and convolution [19,20], the CBAM network proves that the self-attention mechanism can be used as an enhancement of convolution, this paper further proves that convolution can also be used as an enhancement of self-attention, and the effect is good, and based on the idea of U-Net, we design the decoder corresponding to it. Based on the idea of U-Net, a corresponding decoder structure is designed.

Optimizing the representation of deep semantic features of an image is an important topic in the field of vision [21,22]. Some of the work demonstrates that extracting a subset of high-quality features will effectively improve the recognition efficiency of deep networks [23,24], and many works have demonstrated that similar images have the same similar deep semantic features [25,26], and a better encoder can be obtained if the deep semantic feature representation of an image is compressed [27–29]. In the VAD task, compressing the deep semantic feature representation of normal data patterns will increase the boundary of distinction between normal and abnormal data. However, the current work only compresses the channel features of the data, ignoring the spatial structure relationship that exists between neighboring pixel points of the image [25,27–29], we propose a dual-scale deep clustering module that compresses both the channel features and the spatial structure features of the image.

In the VAD task, the memory-like structure can well limit the encoder's representation of anomalous features, and some work has obtained a kind of memory feature vector by weighted averaging the similar feature vectors output from the self-encoder, but such methods require the computation of an additional weight matrix and the structure is completely independent of the encoder, requiring the design of an additional loss function to prevent the memory from learning to degenerate the degradation solution [1,10]. We have implemented a mechanism equivalent to the mnemonic structure using cluster centroids in a two-scale clustering module, which satisfies the need to memorize normal pattern features and further increases the difference between normal and abnormal patterns without introducing additional parameters. In addition, since the clustering module itself is able to guide encoder learning, the generated cluster centroids are always tight and homogeneous, eliminating the need to design complex loss functions and greatly reducing the difficulty of training [27].

In conclusion, the main contributions of this paper are as follows:

- In this paper, we design an efficient and excellent performance video auto-encoder that passes through the input video clips in order to understand the data patterns of normal events in a video scene. The I3D Block is utilized to enhance the learning capability and convergence speed of the ST framework, which outperforms most existing models under large datasets.
- In this paper, we propose a two-scale deep clustering module based on the K-means algorithm which constrains the encoder to generate compact channel features and video spatial features, which helps the model to increase the differentiation between abnormal and normal data patterns. In addition, we use the clustering centroids as memory feature vectors and propose a query mechanism, which can further improve the performance of the network using such feature vectors.
- In this paper, we give an anomaly detection paradigm applied to the video encoder reconstruction task, i.e., the average frame anomaly score of the whole video clip is used to calculate the first frame anomaly score in that video clip. Also this model gives good performance results on the prediction task.

2. Related work

2.1. Anomaly detection based on auto-encoder

Due to the difficulty of labeling all anomalous pattern data in the real world and the different definitions of anomalous pattern data in different scenarios, modeling each anomalous pattern is an unfeasible task. Therefore, the anomaly detection task is usually performed in an unsupervised environment, i.e., the training set contains only normal pattern data [1–3,30]. Based on this, researchers have worked on characterizing normal mode data. Traditional algorithms use Gaussian Mixture Models (GMM) [5], Markov Random Fields (MRF) [4], and Statistical Aggregation Methods [6] to characterize the statistical distribution of normal mode data. Some algorithms in machine learning use 2D wavelet transforms [31] to accomplish visual feature extraction. Whereas most deep learning based methods use auto-encoders to extract feature representations [27,32–37] and most use reconstruction or prediction class methods to learn the normal patterns of the data. Reconstruction class methods use the auto-encoder to extract high-level features of video events and use them to reconstruct the events [7,36,38,39]. 2D convolution auto-encoder methods [1,7,8] stack a number of frames of images on the channel and apply 2D convolution to reduce the video frame size and temporal dimension, extracting the spatial–temporal information to train the network. Nanjun Li et al. further apply 3D convolution methods with a dual-stream network to They further applied 3D convolution method and dual-stream network to train the network by combining RGB video frames with optical flow images [36]. Some researchers have developed a new self-encoder based on the self-attention mechanism and verified that it has excellent results in the task of reconstruction [35,40–42].

Prediction-type methods consider anomalous events to be events that do not meet expectations; these methods predict future frames based only on normal mode historical data, and the testing phase identifies anomalous events by comparing the expected frames with the predicted frames [1,15–17]. Prediction and reconstruction methods can share a network structure, and some methods provide both detection methods [1]. Prediction-based methods are more accurate than reconstruction methods, but require a higher video frame rate as well as more computation. Our method also uses a self-encoder as a backbone network trained on normal mode data to extract normal mode features. We provide both reconstruction and prediction training methods to accommodate different anomaly detection scenarios.

2.2. Video comprehension task

In order to better extract the spatial and temporal information of the video, researchers proposed two mainstream methods of convolution and global attention mechanism for the encoder structure. Andrej Karpathy et al. applied 2D convolution to understand the video, and proposed the classical method of slow fusion of features [12]. Xiaolong Wang et al. introduced self attention mechanism to capture both global and local features of the video [12]. Google team applies Transformer to vision tasks by image chunking and proposes Vision Transformer (VIT) model [41]. Gedas Bertasius et al. propose a VIT-based video encoder architecture and designed a spatial–temporal separated self-attention computation method, which drastically reduces the amount of computation [43]. Sara Atito et al. designed a self-encoder based on

the VIT framework for image restoration tasks [44]. Lee J et al. [45] used masked frame prediction method for predicting the front and back frames of a video under the VIT framework to obtain excellent results. However, the computational price of VIT is expensive, Microsoft Research then published the Swin Transformer method [46] proposed the window self-attention mechanism, which utilizes fixed-size moving window chunks to compute the attention map of the image, significantly reducing the computational overhead, and they then proposed Video Swing Transformer [18]. Jingyun Liang et al. applied the Swin Transformer method to design a self-encoder capable of accomplishing the task of image super-resolution [40].

The basis of our work is based on VST and I3D, and under the influence of the work of Xiaolong Wang et al. we coupled the two models to obtain a video auto-this encoder that has low computational overhead and better captures spatial–temporal global and local feature information. Our coupling adopts the idea of slow fusion, where the feature maps output from I3D are fused with the attention feature maps once at each layer and connected using a residual structure, forcing the network to focus on extracting feature details in a certain region.

2.3. Memory representation of data features

In the anomaly detection task, the boundary between normal mode and abnormal mode data is difficult to distinguish, so there are works devoted to constructing a tight description of the features of normal mode data, which can be divided into two directions, namely, memory structure and deep clustering. Dong Gong et al. [10] introduced a memory structure on the self-encoder framework to save the normal pattern data, such memory structure performs weighted averaging of similar encoded vectors in anticipation of obtaining more representative encoded feature vectors, and the decoding phase uses only the feature vectors saved in the memory to decode. The memory is trained only on normal datasets, and it is no longer trained during the testing phase, which prevents the encoder from reconstructing features that are not memorized by the memory. Hyunjong Park [1] proposes a memory self-encoder based on the U-Net framework, which is designed to further inhibit the expression of anomalous data patterns by jumping over the connectivity layer that connects the decoder to the memory.

Caron et al. [25], proposed an unsupervised method for deep clustering, which feeds the features extracted by the deep network into the K-Means algorithm for clustering, and the clustering centers are designed as the network parameters in order for the network to learn them directly. Yunpeng Chang's team [27,28], designed a clustering module applied to an auto-encoder based on this framework [citing related papers], a structure that allows the auto-encoder to generate more compact feature representations, inversely increasing the boundary between normal and abnormal data patterns.

Although the memory structures and deep clustering methods are fundamentally different in principle, they both aim to characterize a certain class of data. The clustering centroids have been used as a kind of pseudo-labeling, but they can also be regarded as a representative feature representation of the data. Especially in Euclidean space, the mnemonic vectors obtained by weighted averaging are not fundamentally different mathematically from the cluster centroids obtained by gradient optimization. We obtained excellent results in our task using the query mnemonic structure method for clustering centroids, and further propose a two-scale clustering module. This module can guide the encoder to generate more compact feature representations by means of clustering and use the obtained clustered centroids for decoding for better suppression of anomalous pattern representations.

3. Methods

The network framework proposed in this article is summarized in Fig. 1 (Show only reconstruction approach). This article conducts self supervised anomaly detection under the reconstruction and prediction. With the reconstruction approach, the model receives 4 consecutive video frames and outputs the same 4 consecutive video frames, and compares the reconstruction errors among them. In prediction mode, 4 consecutive video frames are received and the fifth frame is predicted. Our model architecture is based on the U-Net [47]. It mainly consists of three components: a video encoder, a deep clustering module, and a video decoder. The video encoder architecture is designed based on Video Swin Transformer [46]. The video decoder removes the skip connection module from the video encoder and introduces a time sampling module to complete the transformation between reconstruction and prediction modes. The video feature deep clustering module adopts the soft interval K-Means clustering method [27,28]. The following text will provide a detailed introduction to the three modules mentioned above.

3.1. Network architecture

3.1.1. Encoder

Our main framework is based on the U-Net model, but in reconstruction and prediction tasks,it is harmful to keep the skip connection, through which the encoder copies the shallow feature information directly to the decoder, this degrades the deep network. We removed this structure and reduced the network layer to 3 layers, replaced the convolution module in the original model with the IST Video encoding module, and replaced the main architecture of the encoder and decoder with the Video Swin Transformer architecture. These methods enable the network to restore the image normally without the skip connection. The IST Video coding block consists of an I3D Block [14] and a Video Swin Transformer Block(ST) [18], both I3D block and ST block have residual structure [48] to prevent the degradation of the model. The parallel connection of I3D block and ST block enables the model to capture both global and local data features, and a new residual structure is formed. We replaced all activation functions in the original framework of these models with GELU functions [49] to prevent a large number of neuronal deaths from affecting the network's ability to extract features.

The video encoder takes as input a video clip with T RGB image frames $X_0 \in \mathbb{R}^{T\times 3\times H\times W}$ to reduce the computational complexity, we retain the ST Framework's patch embed structure $PE(\cdot)$ for initial image encoding, and we use the ith $(i = 1, \ldots, N)$ layer encoding layer as an example to illustrate a complete IST block flow. The input feature is sent into I3D block $Inc_i^e(\cdot)$ and ST block $\psi_i^e(\cdot)$ to extract the deep semantic feature, the superscript e indicates that it belongs to the encoding layer structure. Since the feature map of attention can selectively capture global information, we use the feature map generated by convolution to further enhance the captured information. In the fusion stage, we multiply the feature maps of two different modes and further fuse them using residual structure. Finally, $down_i(\cdot)$ is used to get the input of the next layer. The above procedure can be formalized as:

$$\begin{cases} X_1 = PE\left(X_0\right) \\ X_{i+1} = down\left(Inc_i^e\left(X_i\right)\psi_i^e\left(X_i\right) + \psi_i^e\left(X_i\right) + X_i\right), i = 1, \ldots, N-1 \end{cases} \tag{1}$$

We only do one down-sampling of the time dimension in patch embed stage to reduce computational complexity, and no additional time down-sampling in encoding and decoding stage. This is because too much time down-sampling can improve the reasoning efficiency of the network, but will gradually degrade the deep video encoder into an image encoder, reducing performance. In addition, in practical applications, due to the removal of skip connection layers, the model's reconstruction ability is limited. In order to better balance performance and computational complexity, we only perform spatial downsampling in the first layer, and do not perform spatial downsampling in the following layers.

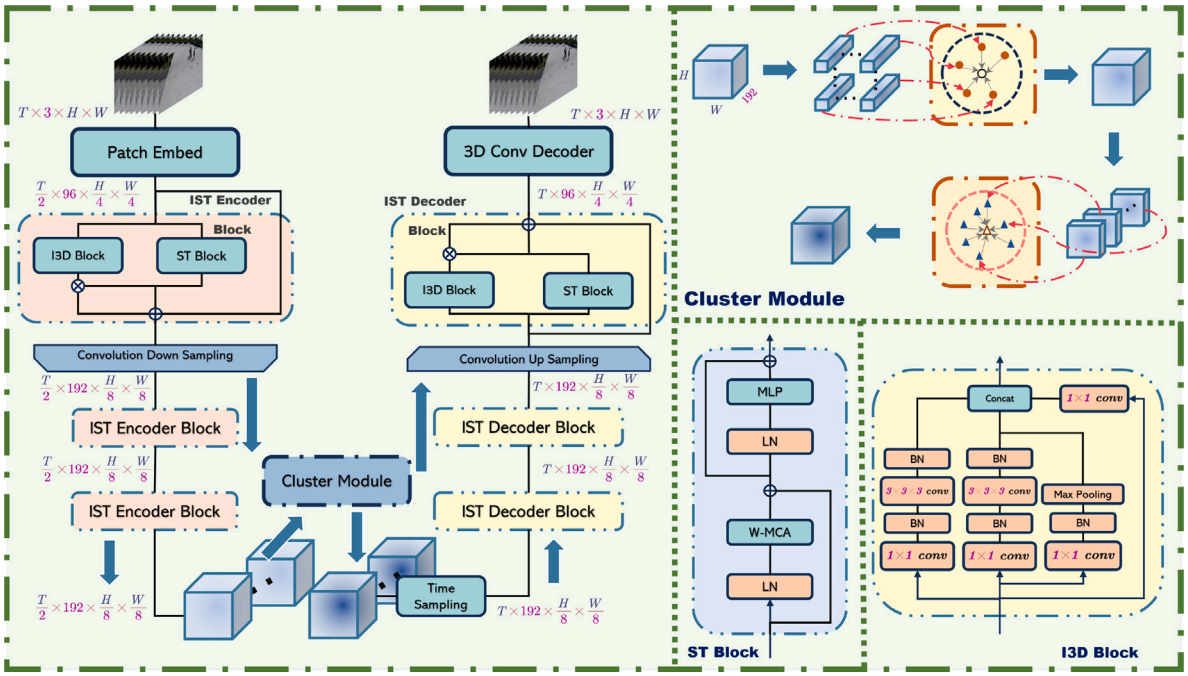


Fig. 1. Overview of the network architecture. (1) The encoder receives a $T \times 3 \times H \times W$ video tensor, which is downsampled by a factor of two in time and a factor of four in space at the patch embed stage. (2) The figure demonstrates how a complete IST block works, where W-MCA is a standard window attention module, and we multiply the adjusted weight feature maps from the ST block with the convolution feature maps output from the I3D block to enhance the features of the convolution feature maps In order to balance the performance with the amount of computation, we perform only one time after the first IST block. Convolution down sampling, which doubles the number of channels and reduces the size of the feature maps, is performed after the first IST block, while the IST block does not change the number of channels. (3) The tensor input to the clustering module will be disassembled into $\frac{T}{2} \times \frac{H}{8} \times \frac{W}{8}$ 192-dimensional feature vectors to be clustered, and then the clustering centroids are used to replace the feature vectors that are close to them and reassembled into the tensor, and for the new generation of the tensor disassembly into 192 clusters, $\frac{T}{2} \times fracT2$ structural feature vectors of $\frac{H}{8} \times \frac{W}{8}$ dimensions are clustered and subsequently reassembled into tensor outputs (4) The decoder finally outputs through a layer of simple 3D convolution network.

3.1.2. Cluster

Since the VAD task is trained only on the normal mode dataset and the scenes are mostly surveillance video footage, there are a large number of redundant data patterns. Compressing the encoder's representation of data features can suppress the encoder's spatial generalization ability to abnormal data patterns without losing encoder performance. As in a deep framework, this compression method should be able to be learned by the network, Inspired by the work of [25–28,50], a dual scale deep clustering module was introduced to achieve this goal.

On the encoding feature scale, suppose K cluster center points are given, then cl_K represents the kth $(1 \le k \le K)$ cluster center, while $CL = \{cl_1, \ldots, cl_K\}$ represents the set of cluster center points. Output video segment features X_N from the N layer video encoder is composed of $T \times H \times W$ C-dimensional feature vectors $\left\{ x_{h,w}^t \middle| x_{h,w}^t \in X_N \right\}$ with $1 \le t \le T$, $1 \le h \le H$, $1 \le w \le W$, which represent the temporal and spatial indexes of video clips, respectively. We will calculate the Euclidean distance between $x_{w,h}^t$ and CL.

We calculate the distance based on the soft allocation technique proposed [28], which enables the encoder to learn more continuous allocation strategies. The calculation formula is as follows:

$$D\left(x_{h,w}^t, CL\right) = \sum_{k=1}^{K} \frac{e^{-\alpha \| x_{h,w}^t - cl_k \|_2}}{\sum_{k=1}^{K} e^{-\alpha \| x_{h,w}^t - cl_k \|_2}} \| x_{h,w}^t - cl_k \|_2^2 \tag{2}$$

where α is an adjustable hyperparameter, Eq. (2) calculates the given eigenvector $x_{w,h}^t$ to each cluster center cl_k The soft allocation distance of can also be understood as a global attention mechanism in Eq. (2).

Inspired by [1,10,13], we find that the clustering centroids obtained by training only on normal data are identical in principle to the memory vectors obtained by weighted averaging of the encoded vectors, i.e., they both find a universal representation of a class of data features. This implies that the clustering centroids can not only assist in training the encoder, but also generate feature maps that are more consistent with the patterns of normal data by querying the clustering centroid vectors, which can further increase the boundary of the distinction between anomalous and normal patterns in reconstructed images. We give this generation method based on the idea of attention mechanism:

$$\hat{x}_{h,w}^t = \frac{e^{-\alpha \| x_{h,w}^t - cl_k \|_2}}{\sum_{k=1}^{K} e^{-\alpha \| x_{h,w}^t - cl_k \|_2}} CL \tag{3}$$

The feature maps $X_{cl} = \left\{ \hat{x}_{h,w}^t, t \in [1,T], h \in [1,H], w \in [1,W] \right\}$ are fed into a spatial clustering module that The module exists C cluster clustering centers, and a set of M cluster points per cluster $SCL = \left\{ scl_1^c, \ldots, scl_M^c \right\}$, the feature map is cut into C clusters to be clustered in the channel direction, with Tfeature subgraph of $H \times W$ dimensions per cluster $\left\{ y_t^c \middle| y_t^c \in X_{cl} \right\}$, where $1 \le c \le C$, represents the cth cluster feature subgraph, similar to Eq. (2) to compute the Euclidean distance from the feature point to the cluster center in each clustered cluster:

$$D\left(y_t^c, scl_m^c\right) = \sum_{m=1}^{M} \frac{e^{-\gamma \| y_t^c - scl_m^c \|_2}}{\sum_{k=1}^{K} e^{-\gamma \| y_t^c - scl_m^c \|_2}} \| y_t^c - scl_m^c \|_2^2 \tag{4}$$

where γ is a hyperparameter, this spatial clustering module clusters the spatial information of the feature map in each feature channel to help the network learn the structured information in the given feature map, but this information comes with extremely strong a priori knowledge, and relying on the spatial clustering points to generate new feature maps will seriously inhibit the generalization ability of the network, so we only perform clustering. Based on the above equation the objective loss function for deep clustering can be defined as:

$$loss_{cl} = \sum_{t=1,h=1,w=1}^{T,H,W} D\left(x_{h,w}^t, CL\right) + \sum_{c=1,t=1}^{C,T} D\left(y_c^t, SCL\right) \tag{5}$$

The initialization of the clustering centroids has a large impact on the performance of the model. But the clustering algorithm in mini-batch training mode can not give a reasonable initialization when the training is turned on, we cut off the gradient flow between the encoder and the clusters at the beginning of the training and make the encoder and the clusters to train on their own for a certain period of time, and then reconnect the gradient flow of the two when the video features have been initially extracted. In the test, the clustering centroids will no longer be learned.

3.1.3. Decoder

The decoder is roughly the same as the encoder in terms of framework, but there are some detailed differences. The input decoder features are dimensionalized once in the timing by $PE(\cdot)$. For the reconstruction task, we found that while this dimensionality reduction can improve the efficiency of the encoding significantly, but the timing of the restoration of the reduced features in the decoding phase will seriously affect the reconstruction performance of the decoder. We chose to temporally sample the restoration of the feature map at the beginning of the decoding phase using an inverse convolution kernel of size $2 \times 1 \times 1$ with a step size of $(2, 1, 1)$. For the prediction task, we use a convolution kernel of size $2 \times 1 \times 1$ with step size $(2, 1, 1)$ to perform temporal downsampling once to obtain the predicted feature map.

Taking the reconstruction task as an example, the decoder receives the features $\hat{X}_N$ output from the Nth layer encoder. Up-sampling the time dimension of them $\varphi(\cdot)$ and then inputs them to the decoding layer for computation, and the complete decoder flow in the reconstructed state can be formally expressed as follows.

$$\begin{cases} \hat{X}_N = \varphi\left(X_N\right) \\ \hat{X}_{i-1} = up\left(Inc_i^d\left(\hat{X}_i\right)\psi_i^d\left(\hat{X}_i\right) + \psi_i^d\left(\hat{X}_i\right) + \psi_i^d\left(\hat{X}_i\right)\right), i = N-1, \ldots, 1 \end{cases} \tag{6}$$

In the decoder output stage, we use a three-layer 3D convolution network for the output, which helps to reconstruct more high-definition images while making the training more stable.

$$\hat{X}_0 = conv\left(\hat{X}_1\right) \tag{7}$$

The decoder flow in the predicted state is similar and we are not going to describe it in detail.

3.2. Loss function

We have two loss functions, reconstruction loss and clustering loss, respectively. We use the L2 paradigm as the reconstruction loss to measure the difference between the input X_0 and the output X_{recon}video clips:

$$loss_{recon} = \|X_0 - \hat{X}_0\|_2 \tag{8}$$

The clustering loss is expressed in Eq. (5), and the final loss function is:

$$loss = loss_{recon} + loss_{cl} \tag{9}$$

3.3. Anomaly score on testing data

In the testing phase of the reconstruction task, we take T video frames as a video clip, but unlike the algorithm using the image auto-encoder, we did not use the error between the reconstructed frame $\hat{x}^t$ and the real frame $x^t (1 \le t \le T)$ to compute the anomaly score for each frame. This is because we found in our experiments that the video encoder produces extremely unstable results when reconstructing consecutive frames, as shown in Fig. 2, where the results of the first and the last frames are more normal, but the intermediate frames have anomalous scores that are too high or too low. This is due to the fact that for the video encoder, the video frames are visible to each other, which produces the phenomenon of temporal event leakage, in order to suppress this phenomenon, we chose to use the average frame anomaly scores of a video clip to calculate the anomaly scores of the first frame of that video clip, which is able to efficiently utilize the information of the video clip. According to [2,7,9], we use the $PSNR$ metric to measure the anomaly score, and the above process can be formally stated as:

$$PSNR\left(X_d, \hat{X}_d\right) = 10\log_{10}\frac{\left(\max\ X_d\right)^2}{\|X_d - \hat{X}_d\|_2} \tag{10}$$

where $1 \le d \le D$, denotes the index of the d frame, and X_d and $\hat{X}_d$ denote the real video clip and reconstructed video clip with the d frame as the first frame, respectively.

In the model evaluation, we follow the work of [15] to slice the long video clip to be detected into D video clips to input the model to get the $PSNR$ value of each frame, which is normalized to the [0, 1] interval to get the final anomaly score. The formula is as follows:

$$S(d) = 1 - \frac{PSNR\left(X_d, \hat{X}_d\right)}{\max_d PSNR\left(X_d, \hat{X}_d\right) - \min_d PSNR\left(X_d, \hat{X}_d\right)} \tag{11}$$

A similar formula is followed for the prediction task, which simply compares the $PSNR$ metrics of the predicted frames.

4. Experiments

4.1. Experiment setup

Datasets. We tested the model on three public benchmark datasets that are widely used in the video anomaly detection domain. (1) Ped2 [51], this dataset includes 16 training videos and 12 test videos with fixed viewpoints on surveillance videos of horizontal sidewalks. The videos have frame-level annotations, and most of the object motion patterns are horizontal, making anomaly detection relatively simple. The anomalous events in the given scenario include cycling, skateboarding, and driving a vehicle on the sidewalk. (2) Avenue [52], which is a comprehensive resource consisting of 16 training videos and 21 test videos on surveillance of complex sidewalks. It uses frame-level and intra-frame mask labeling to detect more complex object motion patterns, spatial perspective motion, and difficult anomalies. It includes 47 anomalous events, such as throwing a bag, approaching or moving away from the camera, and running on the sidewalk. (3)ShanghaiTech [53], which is another resource that can be used for this purpose. The dataset includes 330 training videos and 107 test videos for multi-scene sidewalk intersection monitoring. The videos have frame-level and intra-frame mask annotation, and feature complex object motion patterns and challenging anomaly detection. It contains 330 training videos and 107 test videos on multi-scene sidewalk intersection monitoring with frame-level and intra-frame mask annotation, extremely complex object motion patterns, and high difficulty in event anomaly detection, with a total of 130 anomalous events, such as mobbing, robbing, fighting, etc., which are distributed in 13 different scenes.

Evaluation Metrics. Based on the evaluation metrics widely used in the field of VAD, we use the area under the frame-level curve (AUC) to evaluate the performance of our proposed approach.

Training details. In the training phase, we read in 4 video frames at a time, resizing each image to 224 × 224 and normalizing the pixel values to be within the [0,1] interval. We train our model using the Adam optimizer with L_2 weight-constrained decay and set the decay momentum $\beta = 0.02$. We set the learning rate to 4×10^{-4} and the number of feature clustering points and spatial clustering points to 1024 and 128, respectively. cosine annealing is adopted to gradually decay the learning rate. We set the depth of each layer in the SW block $N$to 6.

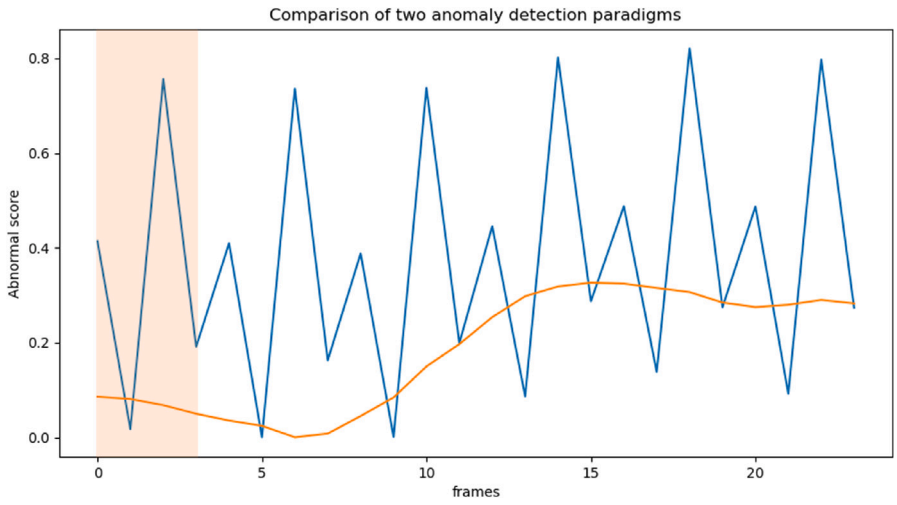


Fig. 2. We compare the results of two detection paradigms, the frame-by-frame detection paradigm (blue line in the figure) and the frame average detection paradigm for video clips (yellow line part in the figure). In the frame-by-frame detection paradigm, we test the already trained network with 4 frames as a video clip, using a paradigm that directly calculates the anomaly score value for each frame, and obtain crashworthiness results. As can be seen in the figure, in a complete video clip of the input (the orange part of the figure), the first and the last frames have more normal anomaly scores, but the middle frames have too high or too low anomaly scores, and the overall is extremely unstable. The video clip frame average detection paradigm, on the other hand, gives stable and more trending results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We run 60, 60, 80 epochs on Ped2, Avenue, ShanghaiTech datasets, respectively. batch-size of each dataset is set to 20, and the BN layer is frozen uniformly during training. We train on four **NVIDIA GTX2080TI GPUs**.

4.2. Experiment result

We test the model under three datasets. From Table 1 we can see that our method achieves leading results on the Shanghai Tech dataset, with a substantial improvement over the state-of-the-art method based on memory modules [1,10] with 3.2%, 2.5% respectively. And compared to the state-of-the-art method that also has a clustering structure and uses RGB difference maps [27,28] with a small improvement. Meanwhile, our method has a substantial improvement compared to both methods using 2D convolution [7,38,39] and have 12.8%, 5.8%,3.0% improvement on ShanghaiTech dataset, respectively. The performance on Ped2 is poorer, but this is understandable; the Ped2 dataset is small, and our network employs a complex mechanism that mixes convolution and self-attention, requiring enough data for learning to find general behavioral patterns. The fact that our method achieves a better level of performance on the larger Avenue dataset also demonstrates that our method requires sufficiently large data for training. In addition our method did not use aids such as optical flow maps [17,36], RGB difference maps [27,28] for training, and achieved good results with only the deep clustering module to assist in training, which proves the effectiveness of our proposed model.

Moreover, our proposed reconstruction paradigm also outperforms other methods using reconstruction paradigms on the ShanhaiTech dataset [7,36,38,39], which suggests that our reconstruction paradigm is more suitable for video encoders.

In Fig. 3 we present the image prediction effect of our model on three datasets Ped2, Avenue, and ShanghaiTech, we can see that our model is able to detect the region where the anomaly occurs well, and in the predicted state, the anomalous objects show obvious blurring and lack of details, and in Fig. 5 we can see that the scores appear to be significantly increased when the anomaly occurs, and the scores are maintained at a lower level in the anomalous unoccurring region, which further shows that our model is able to judge the boundaries of anomalous occurrences very well.

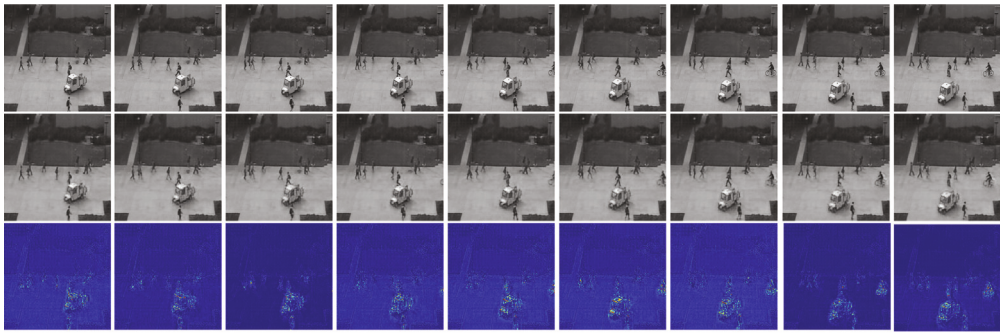
Table 1
AUC of different methods on the Ped2, Avenue and ShanghaiTech datasets. Numbers in bold indicate best performance, underlined numbers indicate next best performance.

Methods	Ped2	Avenue	ShanghaiTech
Conv2D-AE [7]	85.0%	80.0%	60.9%
Conv3D-AE [9]	91.2%	77.1%	–
ConvLSTM-AE [8]	88.1%	77.0%	–
MemAE [10]	94.1%	83.3%	71.2%
TSC [38]	91.0%	80.6%	67.9%
ST-CaAe [36]	92.9%	83.5%	–
Abati [32]	95.41%	–	72.5%
FFP [15]	95.4%	84.9%	72.8%
MNAD-Recon [1]	90.2%	82.8%	69.8%
MNAD-Pred [1]	97.0%	88.5%	70.7%
Zhong et al. [39]	**97.7**%	**88.9**%	70.7%
CCDA [27]	96.5%	86.0%	73.3%
AnoPCN [16]	96.8%	86.2%	73.6%
AMMC-Net [17]	96.9%	86.6%	73.7%
Xie et al. [28]	96.7%	87.1%	73.7%
Ours-Pred	92.2%	86.2%	**73.76**%
Ours-Recon	90.8%	83.1%	73.3%

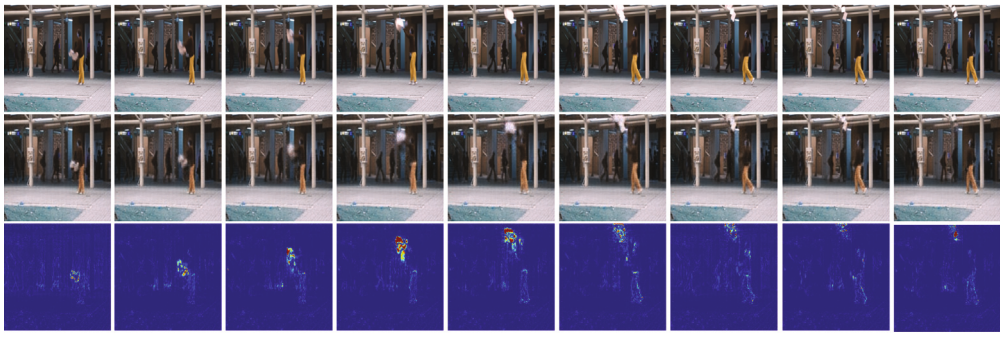
4.3. Ablation study and analysis

In this section, we conduct several ablation studies focusing on the effects of the I3D Block, the ST Block, and the two-scale clustering block on the model performance. With keeping the rest of the settings unchanged, the test results are shown in Table 2, where we can see that the I3D Block has a significant performance enhancement on the self-attention mechanism, which is 2.2% and 3.5% compared to using only the ST Block with the I3D Block, respectively. In addition feature clustering also has a significant improvement on the model performance with a 0.8% performance enhancement, and the spatial clustering module improves the performance of the model by 0.4%. Here we need to emphasize that using only Swin Transformer with the feature clustering module with class memory structure will result in training failure. We believe that this is due to the fact that the normality of the initialization of the clustering centroids corrupts the feature weight maps tuned by the self-attention mechanism. The model is trained successfully after we discard the step of Eq. (3).

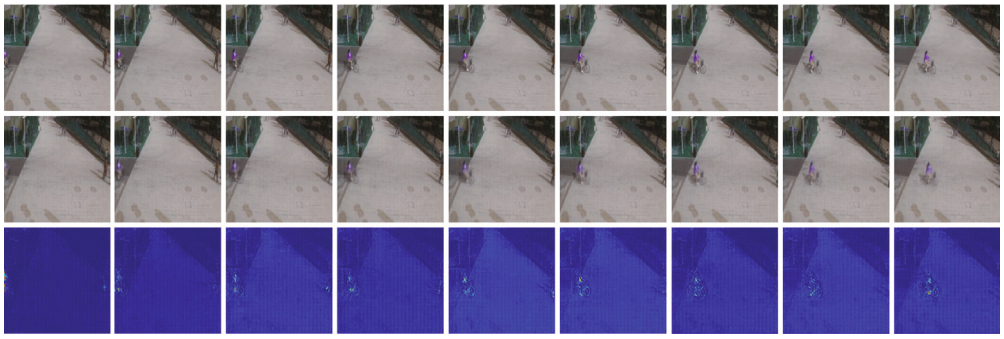
In addition if we combine Table 2 with Table 2, our method without the clustering module is also far better than the method using a regular



(a) Ped2

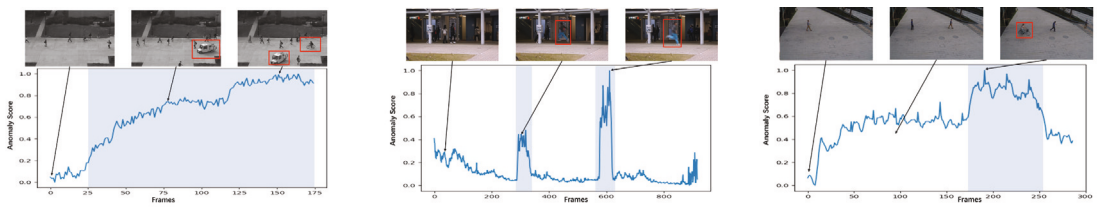


(b) Avenue



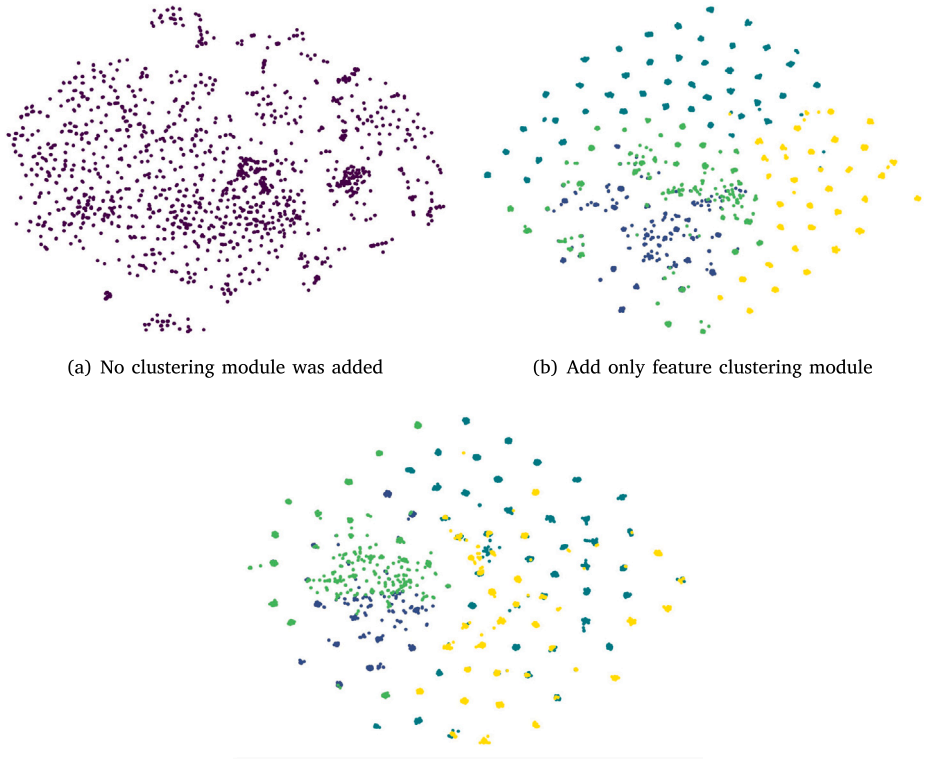
(c) ShanghaiTech

Fig. 3. Examples of video frames prediction on three datasets. The first row shows the input video frames, the second row shows the reconstructed frames, and the third row shows the visualization of the attention map in the jet color map. The higher attention weight area is represented closer to red while the lower area is represented closer to blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Ped2 (b) Avenue (c) ShanghaiTech

Fig. 4. Anomaly score curves of several test samples of our method on three benchmark datasets.



(a) No clustering module was added

(b) Add only feature clustering module

(c) Use a two-scale clustering module

Fig. 5. Anomaly score curves of several test samples of our method on three benchmark datasets. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Evaluation of different components of our model on the ShanghaiTech dataset. Results show that the combination of all components gives the best performance.

ST block	✓	–	✓	✓	✓
I3D block	–	✓	✓	✓	✓
Feature cluster	–	–	–	✓	✓
Space cluster	–	–	–	–	✓
AUC	69.9%	68.9%	72.5%	73.3%	73.7%

Table 3
AUC of the proposed method with different **feature cluster centroid** numbers on the ShanghaiTech dataset.

Numbers	64	128	512	1024
AUC	66.1%	70.1%	72.4%	73.7%

Table 4
AUC of the proposed method with different **space cluster centroid** numbers on the ShanghaiTech dataset.

Numbers	16	32	64	128
AUC	72.9%	73.1%	73.71%	73.76%

image encoder [7,38,39], with 11.6%, 4.6%, and 1.8% improvement, respectively.

In Tables 3 and 4, we fix other parameters and only change the number of clustering centroids to explore its effect on model performance, Table 3 shows that when the number of feature clustering points is small, it will drag down the encoding ability of the model, and the model performance improves with the number of feature clustering centroids, which is consistent with the conclusion of the size of the memory in the memory structure. In Table 4, when the number of spatial clustering points is greater than 32, the ability to enhance the model performance is limited.

We use the T-SNE [54] to visualize 10,000 randomly selected coded features for dimensionality reduction, using the clustered points closest to the feature points as pseudo-labels, and the results are displayed in Fig. 5, where similar features are automatically grouped together due to the use of an attentional mechanism, whereas the features employing biscale clustering are more uniformly distributed in the coding space into a number of clustered clusters. In addition, feature points using only single-scale clustering are well categorized into several clusters of features, whereas features using double-scale clustering cluster the clustered points themselves one more time in the spatial dimension (there is an overlap between the yellow and dark green features in the figure).

4.4. Discussion

Our approach demonstrates that the convolution module provides a performance enhancement to the self-attention framework, but does not actually demonstrate how this enhancement works. We believe that the feature maps tuned by the self-attention mechanism perhaps perform a weight assignment for the feature maps of the convolutional outputs, which can add some feature details to the parts of the network that are of interest and thus enhance the feature extraction ability of the network, but we did not find tools that can visualize multi-dimensional feature maps to observe such phenomena. If this phenomenon can be observed or proved, it may bring good development for the work related to unified convolution and self-attention mechanism. In addition, we provide a new training method for memory modules from a clustering perspective. This method can update the memory by back-propagation without modifying the memory individually in the training mode, and at the same time can constrain the training of the encoder,

which we believe can provide a more effective research idea for the auto-encoder of the memory-guided class.

5. Conclusion

In this paper, we give a convolution-enhanced self-attentive video auto-encoder based on the U-Net architecture to better characterize the spatial–temporal features of the video data, specifically, in order to make the encoder have the performance of capturing the global information capability without losing the ability to describe the local information capability, we augmented the feature maps of the output of the ST Block by an I3D block, and the improvement has achieved excellent results in the experiments achieved excellent results. In addition, we redesigned a dual-scale feature clustering module on the framework of the deep K-means algorithm in order to better compress the feature representation of normal data to increase the distinction boundary between normal and abnormal data. This module clusters the data features once from the feature scale and the spatial structure scale respectively, and outputs a more compact feature map by querying the cluster centroids in the feature clustering. We also show that the cluster centroids in feature clustering are functionally equivalent to a class of memory structures, but at the same time are better able to assist in training auto-encoders. In the reconstruction task, in order to address the temporal event leakage problem of the video encoder, we propose an anomaly score computation paradigm for the video encoder that utilizes the average frame anomaly scores of the entire video clip to compute the anomaly score of the first frame in that video clip. Extensive experiments on three benchmark datasets show that our method outperforms most existing methods on large datasets with complex patterns.

CRediT authorship contribution statement

Shaoming Qiu: Supervision, Software, Resources, Project administration, Formal analysis. **Jingfeng Ye:** Writing – review & editing, Writing – original draft, Validation, Software, Resources, Project administration, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] H. Park, J. Noh, B. Ham, Learning memory-guided normality for anomaly detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14372–14381.

[2] B. Ramachandra, M.J. Jones, R.R. Vatsavai, A survey of single-scene video anomaly detection, IEEE Trans. Pattern Anal. Mach. Intell. 44 (5) (2020) 2293–2312.

[3] R. Nayak, U.C. Pati, S.K. Das, A comprehensive review on deep learning-based methods for video anomaly detection, Image Vis. Comput. 106 (2021) 104078.

[4] H. Nallaivarothayan, C. Fookes, S. Denman, S. Sridharan, An MRF based abnormal event detection approach using motion and appearance features, in: 2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, IEEE, pp. 343–348.

[5] S.R. Bond, A. Hoeffler, J.R. Temple, GMM estimation of empirical growth models, 2001, Available at SSRN 290522.

[6] V. Saligrama, Z. Chen, Video anomaly detection based on local statistical aggregates, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 2112–2119.

[7] M. Hasan, J. Choi, J. Neumann, A.K. Roy-Chowdhury, L.S. Davis, Learning temporal regularity in video sequences, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 733–742.

[8] W. Luo, W. Liu, S. Gao, Remembering history with convolutional lstm for anomaly detection, in: 2017 IEEE International Conference on Multimedia and Expo, ICME, IEEE, pp. 439–444.

[9] A. Zimek, E. Schubert, H.-P. Kriegel, A survey on unsupervised outlier detection in high-dimensional numerical data, Stat. Anal. Data Min.: ASA Data Sci. J. 5 (5) (2012) 363–387.

[10] D. Gong, L. Liu, V. Le, B. Saha, M.R. Mansour, S. Venkatesh, A.v.d. Hengel, Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1705–1714.

[11] T.-N. Nguyen, J. Meunier, Anomaly detection in video sequence with appearance-motion correspondence, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1273–1283.

[12] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732.

[13] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794–7803.

[14] J. Carreira, A. Zisserman, Quo vadis, action recognition? a new model and the kinetics dataset, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6299–6308.

[15] W. Liu, W. Luo, D. Lian, S. Gao, Future frame prediction for anomaly detection–a new baseline, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6536–6545.

[16] M. Ye, X. Peng, W. Gan, W. Wu, Y. Qiao, Anopcn: Video anomaly detection via deep predictive coding network, in: Proceedings of the 27th ACM International Conference on Multimedia, pp. 1805–1813.

[17] R. Cai, H. Zhang, W. Liu, S. Gao, Z. Hao, Appearance-motion memory consistency network for video anomaly detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, pp. 938–946.

[18] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, H. Hu, Video swin transformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3202–3211.

[19] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, in: Proceedings of the European Conference on Computer Vision, ECCV, pp. 3–19.

[20] X. Pan, C. Ge, R. Lu, S. Song, G. Chen, Z. Huang, G. Huang, On the integration of self-attention and convolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 815–825.

[21] C. Zhang, H. Fu, S. Liu, G. Liu, X. Cao, Low-rank tensor constrained multiview subspace clustering, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1582–1590.

[22] V. Noroozi, S. Bahaadini, L. Zheng, S. Xie, W. Shao, S.Y. Philip, Semi-supervised deep representation learning for multi-view problems, in: 2018 IEEE International Conference on Big Data, Big Data, IEEE, 2018, pp. 56–64.

[23] İ. Yağ, A. Altan, Artificial intelligence-based robust hybrid algorithm design and implementation for real-time detection of plant diseases in agricultural environments, Biology 11 (12) (2022) 1732.

[24] Y.B. Özçelik, A. Altan, A comparative analysis of artificial intelligence optimization algorithms for the selection of entropy-based features in the early detection of epileptic seizures, in: 2023 14th International Conference on Electrical and Electronics Engineering, ELECO, IEEE, 2023, pp. 1–5.

[25] M. Caron, P. Bojanowski, A. Joulin, M. Douze, Deep clustering for unsupervised learning of visual features, in: Proceedings of the European Conference on Computer Vision, ECCV, pp. 132–149.

[26] D. Bo, X. Wang, C. Shi, M. Zhu, E. Lu, P. Cui, Structural deep clustering network, in: Proceedings of the Web Conference 2020, pp. 1400–1410.

[27] Y. Chang, Z. Tu, W. Xie, J. Yuan, Clustering driven deep autoencoder for video anomaly detection, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16, Springer, pp. 329–345.

[28] Y. Chang, Z. Tu, W. Xie, B. Luo, S. Zhang, H. Sui, J. Yuan, Video anomaly detection with spatio-temporal dissociation, Pattern Recognit. 122 (2022) 108213.

[29] M.Z. Zaheer, A. Mahmood, M. Astrid, S.-I. Lee, Clustering aided weakly supervised training to detect anomalous events in surveillance videos, IEEE Trans. Neural Netw. Learn. Syst. (2023).

[30] L. Alzubaidi, J. Bai, A. Al-Sabaawi, J. Santamaría, A. Albahri, B.S.N. Al-dabbagh, M.A. Fadhel, M. Manoufali, J. Zhang, A.H. Al-Timemy, et al., A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications, J. Big Data 10 (1) (2023) 46.

[31] Y.B. Özçelik, A. Altan, Overcoming nonlinear dynamics in diabetic retinopathy classification: a robust AI-based model with chaotic swarm intelligence optimization and recurrent long short-term memory, Fractal Fract. 7 (8) (2023) 598.

[32] D. Abati, A. Porrello, S. Calderara, R. Cucchiara, Latent space autoregression for novelty detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 481–490.

[33] Z. Yang, J. Liu, P. Wu, Bidirectional retrospective generation adversarial network for anomaly detection in videos, IEEE Access 9 (2021) 107842–107857.
[34] L. Wang, J. Tian, S. Zhou, H. Shi, G. Hua, Memory-augmented appearance-motion network for video anomaly detection, Pattern Recognit. 138 (2023) 109335, https://github.com/Owen-Tian/MAAM-Net.
[35] J. Lee, W.-J. Nam, S.-W. Lee, Multi-contextual predictions with vision transformer for video anomaly detection, in: 2022 26th International Conference on Pattern Recognition, ICPR, IEEE, pp. 1012–1018.
[36] N. Li, F. Chang, C. Liu, Spatial-temporal cascade autoencoder for video anomaly detection in crowded scenes, IEEE Trans. Multimed. 23 (2020) 203–215.
[37] Y. Zhao, B. Deng, C. Shen, Y. Liu, H. Lu, X.-S. Hua, Spatio-temporal autoencoder for video anomaly detection, in: Proceedings of the 25th ACM International Conference on Multimedia, pp. 1933–1941.
[38] W. Luo, W. Liu, S. Gao, A revisit of sparse coding based anomaly detection in stacked rnn framework, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 341–349.
[39] Y. Zhong, X. Chen, J. Jiang, F. Ren, A cascade reconstruction model with generalization ability evaluation for anomaly detection in videos, Pattern Recognit. 122 (2022) 108336.
[40] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, R. Timofte, Swinir: Image restoration using swin transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1833–1844.
[41] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv preprint arXiv:2010.11929.
[42] G. Li, G. Cai, X. Zeng, R. Zhao, Scale-aware spatio-temporal relation learning for video anomaly detection, in: European Conference on Computer Vision, Springer, pp. 333–350.
[43] G. Bertasius, H. Wang, L. Torresani, Is space-time attention all you need for video understanding? in: ICML, Vol. 2, p. 4.
[44] S. Atito, M. Awais, J. Kittler, Sit: Self-supervised vision transformer, 2021, arXiv preprint arXiv:2104.03602.
[45] J.-C. Feng, F.-T. Hong, W.-S. Zheng, Mist: Multiple instance self-training framework for video anomaly detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14009–14018.
[46] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.
[47] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, Springer, pp. 234–241.
[48] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
[49] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), 2016, arXiv preprint arXiv:1606.08415.
[50] M. Ronen, S.E. Finder, O. Freifeld, Deepdpm: Deep clustering with an unknown number of clusters, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9861–9870.
[51] W. Li, V. Mahadevan, N. Vasconcelos, Anomaly detection and localization in crowded scenes, IEEE Trans. Pattern Anal. Mach. Intell. 36 (1) (2013) 18–32.
[52] C. Lu, J. Shi, J. Jia, Abnormal event detection at 150 fps in matlab, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 2720–2727.
[53] Y. Zhang, D. Zhou, S. Chen, S. Gao, Y. Ma, Single-image crowd counting via multi-column convolutional neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 589–597.
[54] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).

Shaoming Qiu, Male, Associate Professor, Master Supervisor, Member of CCF Specialized Committee, Main Research Fields: Fault Diagnosis, Command and Control, Intelligent Algorithm.

Jingfen Ye is currently studying for a Master's degree at Dalian University, and received his Bachelor of Science degree from the School of Mathematics and Statistics, Minnan Normal University, in 2021. His research interests mainly include computer vision and natural language processing.

Jiancheng Zhao received the B.S. degree in Internet of Things from Changshu Institute of Technology, Suzhou, China, in 2021. He is currently pursuing the master's degree with the Communication and Network Laboratory, Dalian University. His current research interests are edge computing and Digital.

Lei He is a current Master of Science student at Dalian University and received his Bachelor of Science degree from the College of Science, Jiangsu Ocean University, Jiangsu, China, in 2021, and his research interest is in computer vision.

Liangyu Liu is currently a master student at Dalian University and received his B.E. degree in Computer Science and Technology from Chaohu College in 2020. His research interests mainly include fault diagnosis and command and control.

Bicong E. is currently working on his master's degree at Dalian University and received his bachelor's degree in Vehicle Engineering from Southwest University of China in 2019. His research interests mainly include fault diagnosis and software defect prediction.

Xinchen Huang He is currently a Master student at Dalian University and received his degree in Computer Science and Technology from Anhui Agricultural University in 2021. His research interests mainly include knowledge graph and assisted decision making.