COVAD: Content-oriented video anomaly detection using a self-attention based deep learning model

Wenhao Shao, Praboda Rajapaksha, Yanyan Wei, Dun Li, Noel Crespi, Zhigang Luo

• Article •

COVAD: Content-Oriented Video Anomaly Detection using a Self-Attention based Deep Learning Model

Wenhao Shao [1,2*], Praboda Rajapaksha [2], Yanyan Wei [3], Dun Li [2], Noel Crespi [2], Zhigang Luo [1]

1. *College of computer, National University of Defense Technology, Changsha* 410073*, CHIINA*
2. *Telecom SudParis, IMT, Institut Polytechnique de Paris, 91764 Palaiseau, FRANCE*
3. *Zhengzhou College of Finance and Economics, 450000 Zhengzhou CHINA*

*** Corresponding author**， shaowenhao007@gmail.com

Received:

Abstract Background Video anomaly detection has always been a hot topic and attracting an increasing amount of attention. Much of the existing methods on video anomaly detection depend on processing the entire video rather than considering only the significant context. This paper proposes a novel video anomaly detection method named COVAD, which mainly focuses on the region of interest in the video instead of the entire video. Our proposed COVAD method is based on an auto-encoded convolutional neural network and coordinated attention mechanism, which can effectively capture meaningful objects in the video and dependencies between different objects. Relying on the existing memory-guided video frame prediction network, our algorithm can more effectively predict the future motion and appearance of objects in the video. Our proposed algorithm obtained better experimental results on multiple data sets and outperformed the baseline models considered in our analysis. At the same time we improve a visual test that can provide pixel-level anomaly explanations.

Keywords Video Surveillance; Video Anomaly Detection; Machine Learning; Deep Learning; Neural Network; Coordinate attention

1 Introduction

Video anomaly detection is a research hotspot in the field of computer vision, attracting many researchers [1,2,3]. With the improvement of hardware processing performance and the increase of human resource costs, it becomes more unreasonable to consider manual 24-hour uninterrupted video monitoring approaches. The Vincent's SmartCatch intelligent video surveillance systems operated at the San Francisco International Airport, Helsinki airport and several other airports are capable of detecting physical security breaches real-time. After analysis the statistics from these systems, airport leaders found hundreds of incidents that

threaten security every day, which have not been discovered before [4] with unsustainable human monitoring systems. Therefore, it is imminent to propose an effective intelligent video anomaly detection technology that can further detect video anomalies in real scenarios. The core of video anomaly detection technology is to find out abnormal events from a series of continuous videos. However, in the real world, abnormal events cannot define accurately and have no boundary [5]. Therefore, it is impossible to label all abnormal events to generate datasets to train supervised models. In addition, it is difficult to collect sufficient number of types and quantities of abnormal data and hence, it is not reasonable to use supervised learning algorithms for video anomaly detection tasks. Existing algorithms are mostly unsupervised and semi-supervised [6].

Many videos anomaly detection algorithm uses convolutional neural network (CNN) to learn video features, including temporal dimension features and spatial dimension features. Then, use inverse coding to reconstruct the video or combine with optical flow technology to predict the next frame. According to the definition of training loss, existing unsupervised and semi-supervised video anomaly detection algorithms are divided into two categories, one is reconstruction-based anomaly detection [7,8,9], and the other is prediction-based [10,11,12] anomaly detection algorithms. The reconstruction-based anomaly detection algorithm defines the reconstruction loss as the training loss. Reconstruction-based method assumes that the detection model is trained by a large amount of normal data, the model can accurately describe normal events, extract video features, and restore video features to video frames with small reconstruction errors. If no data objects participate in the training especially for abnormal events, then the model will get a large loss when reconstructing abnormal videos. In the detection phase, error thresholds are set to detect abnormal events. To future frame prediction, the training error of the prediction-based video anomaly detection algorithm is the prediction error, and the basic structure is to extract the video features of the previous frames and predict the features of the future frames. During the training phase, the loss between the predicted future frame and the real future frame is calculated, and the network parameters are updated. This paper proposes a video anomaly detection algorithm for future frame prediction and thus, it follows the assumptions that the models trained on normal data sets have small errors in predicting future frames of normal events, and abnormal events have higher prediction errors due to their uncertainty [13].

After the emergence of deep learning techniques, the use of CNN to extract video features instead of the original hand-made features greatly saves time and cost, and achieved higher accuracy after training the models on specific scenario. The basic structure of current video anomaly detection algorithms is almost the same. It is mainly divided into the following steps: input the video frame into the encoder to extract the features using the training method of the adversarial CNN, and use the decoder to restore the features. Then, calculate the error between restored features and the original features, and adjust the network parameters to make the extracted features closer to the video frame. The neural network has strong representation ability, but in order to prevent unbounded expression, it is necessary to limit the representation ability of the neural network by adjusting the pooling part of the network structure. In addition, it is difficult to obtain an accurate model to discriminate anomalies with a single network structure parameter training and thus, it is necessary to record the extracted video frame features (all training sets are from normal events). One of the

most typical solutions for this is to adapt memory-guided video anomaly detection algorithm as proposed in [14] in 2020. This method adopts the latest U-Net symmetric network, which has strong representation ability. In U-Net network, the back-sampling technology in the decoder can make up for the loss of spatial information in the pooling process and the memory storage module further retains the features and feeds it back to the decoder for preserving spatial information.

Video anomaly detection is different from traditional video analysis. Usually, abnormal events only occur in a small part of the video pixels and therefore, are inappropriate to focus on all video pixels as most of the video pixels are harmless, or called the background. Therefore, in the process of video feature extraction, attention should be focused on a few detectable partial objects. Object detection is very complected, which will consume a lot of time during video processing. Therefore, it is not advisable to use object detection in the training phase to focus attention on anomalous parts.

In this paper, a content-based video anomaly detection algorithm - COVAD, is proposed and its network structure is modified based on the original memory-based video anomaly detection algorithm. The main goal of optimization in the training network is to focus on the objects in the video frame. We use content-based attention mechanism to optimize the structure of the encoding network and removed the last batch of normalization layer of the U-Net network. The former is used to focus on the target or content in the video and the latter is used to limit the powerful bias of the neural network as it is important to blur the boundary between normal data and abnormal data in Powerful representations. Compared with the object detection algorithm, the attention mechanism is lightweight, does not take up a lot of time, and can effectively process video. The memory storage module stores more important content information, rather than the entire video frame pixels. Our experiments are deployed on the USCD [15] and Avenue datasets [16], and the experimental results show that the algorithm proposed in this paper has better results compared to the bench mark models.

The main contributions of this paper are 1) to propose a novel video anomaly detection method, called COVAD, for future frame prediction by combining the content-based attention mechanism, which can resist the interference of noise and focus on extracting the features of objects in the video, 2) to redefine memory module, which is used to classify and memorize various normal behavioral patterns available in video streams, and 3) to further improve the performances of video anomaly detection models focused on both normal and exceptional events. The experimental results show that the performance of the proposed COVAD algorithm in this paper is significantly higher than that of the baseline models considered in this work.

2 Related Work

Before the advent of deep neural networks, video anomaly detection techniques usually employed handcrafted video appearance and motion features, statistics, regression, hashing, and classification. Venkatesh et al. [17] presented the insight that if abnormal behavior is local, then even normal events exhibit dependencies, and the optimal rules for normal behavior should also be local. This paper proposed a probabilistic framework to detect abnormal events in videos, and calculated a comprehensive score for each

video segment through local experience and local statistics to detect abnormal events. In 2015, Kai-Wen et al. [18] proposed a hierarchical framework, which treated the process of detecting anomalies as a 3D pattern matching problem and detected abnormal events through hierarchical and Gaussian regression. In 2016, Ying Zhang et al. [19] proposed using a locality-sensitive hash filter to detect abnormal events. This method hashes the normal data set in the bucket through the locality-sensitive hash function, and matches the coordinate points in the bucket in the detection stage. Detect anomalies. In 2016, Mahmudul Hasan et al. combined hand-crafted features and auto-encoding techniques to propose an end-to-end learning framework that uses a fully convolutional feed-forward auto-encoder to learn features and classifiers, trained from multiple mixed data Model [20].

However, these methods usually require manual extraction of video features, which consumes a lot of time and labor costs, and identification is much more difficult in real-time detection. After the advent of deep neural networks, the situation has improved, and researchers and scholars have delivered the work of feature extraction to neural networks, and the benefits of neural networks are much greater than manual feature extraction [20]. Such similar classic algorithm was introduced by Weixin Luo et al. in 2017 CVPR [21], which proposed that the features extracted by deep neural networks are more accurate than traditional hand-made features, and proposed an approach that combined LSTM and auto-encoding techniques to extract the appearance features and motion features of videos. Compared with [20], the features extracted by this method are more accurate, efficient, and have better experimental results.

In the previous research works on anomaly detection algorithms considered that the reduction of reconstruction error as an objective function of the mainstream solution (flagged as an anomalous event). However, there is a problem with this method, in which, the entire training process only reduced the reconstruction error of normal events, and there is no guarantee that anomaly detection has considerable error, and abnormal events may still be reconstructed. Therefore, in 2018, Wen Liu et al. proposed a video anomaly detection framework for future frame prediction [10]. During the training phase of their model, the first $n-1$ frames of the video frame sequences are used as inputs, and the nth frame is considered as predicted. Loss function in their method was defined as the error between the predicted nth frame and the real nth frame. The experimental results of this method shown that, it was not reduced the reconstruction error as the objective function. The theoretical basis and assumption of this method was that the abnormal events mostly occur suddenly. When the frames belong to the normal events are used as input, the motion trajectory or appearance features of future frames will change and has been limited to a certain range. Once the error between the predicted future frame and the real future frame exceeded the given range, the video frame sequence is likely to be abnormal. The future frame prediction scheme successfully overcomes the problems of previous reconstruction-based methods. Furthermore, there is another improvement measure. For example, in 2019, Dong Gong proposed a deep automatic coding anomaly detection algorithm for memory storage aggregation [22], which proposed that due to the excellent representation ability of neural networks, the reconstruction error of abnormal events is not always greater than the threshold. As a result, their article proposed to add a memory storage module to improve the fitting ability of the model and normal events and expanded the gap between abnormal events. Their proposed model improved the

detection ability of anomaly events. In 2020, Hyunjong Park et al. [14] optimized on the basis of [22], combined with U-Net network to further limit the expressive ability of neural network, and proposed future frame prediction and reconstruction-based video anomaly detection algorithm. This method saved time and cost, and further improved the detection accuracy of abnormal events. Although there are still some technical improvements over some other proposed models in the state-of-the-art, they are obviously not reasonable improvements in terms of theoretical and experimental results. An article published at the CVPR conference [23] in 2021 proposed the use of multi-task learning and pseudo-label generation to solve the problem of uneven distribution of normal events. In another study published on CVPR [24] in 2021 proposed a novel content-oriented lightweight attention mechanism network, which focused on the network training on the content of the video frame. These two studies can be considered as novel improvements in video anomaly detection methods. In addition, there are some additional tricky new architectures, which will be described in the conclusion section of this article.

3 Methodology

This paper focuses on combining memory module guidance and the content-based attention mechanism to propose a new video anomaly detection algorithm, which is mainly based on future frame prediction. The COVAD method proposed in this paper first, learns the temporal and spatial features of the video and then, maps its features to the memory storage module and updates the records of the memory storage module. Finally, the decoder network is used to restore video features, and calculate the difference between the predicted video frame and the real video frame, and evaluate the error. However, unlike previous methods, this paper modifies both the encoder and decoder network, and proposes a content-oriented self-attention mechanism by integrating encoder/decoder network mainly analysis the video content using the features learned from the neural network. Figure 1 depicts the COVAD system architecture and more details about this system will provide in detail in the following sections.

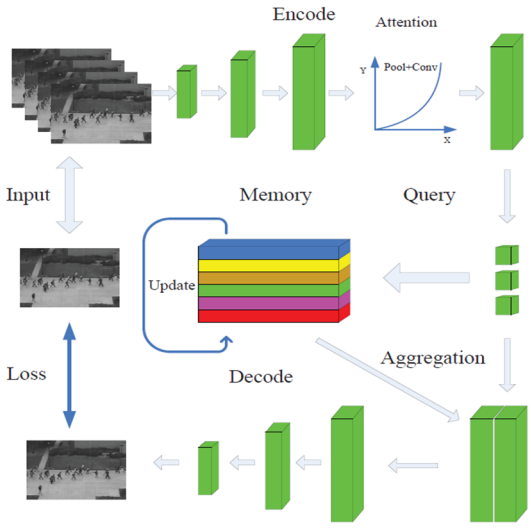


Figure 1 The Algorithm Framework: 1. Extract video features through an encoder, 2. Then input collaborative attention mechanism to redistribute weights, 3. Read memory module and update, 4. Restore the aggregated query features and memory module features to video frames, 5. Calculate the loss, backpropagate, and update parameters

The area where abnormal events occur in a video only occupies a small part of the entire video frame, and therefore, most of the scenes in video frames are useless for detecting abnormal events, which we call in this research as the background. In video anomaly detection, it is generally accepted that stereoscopic, interdependent content, or objects in the video are more worthy of attention. However, most algorithms today are not designed with this argument in mind. Therefore, motivated by this, our paper proposes a novel video anomaly detection algorithm that incorporates a state-of-the-art content-oriented self-attention mechanism to training on the important content of video frames, rather than the providing much attention to the background.

The algorithm proposed in this paper is mainly divided into three parts: the encoder, the memory storage module, and the decoder.

- The encoder is used to extract the temporal and spatial features of the video,
- The memory storage module records the behavior patterns of normal events
- The encoder restores the extracted features as video frames.

Encoder and decoder: The most popular encoder and decoder used for video processing at present is the U-Net symmetric network. The structure of the network is symmetrically distributed, which can effectively represent the process of feature extraction and feature restoration of video frames, as shown in Figure 1. Apart from that, due to the special aggregation mode of the U-Net network (meaningful data is appended when restoring features) and the up-sampling process, the motion and appearance information of the video can be preserved to the greatest extent.

Memory storage module: This module is a sparse binary matrix, which is updated during the training process, and constantly fits the behavioral patterns of normal events to realize the function of memorizing normal behavioral patterns. The basic principle is to use a sparse binary matrix to record the video features in each iteration. As the number of iterations increases, the sparse matrix of the memory module fits the normal behavior pattern during training.

In our proposed approach, model input which is the continuous video frame sequence $Seq = \{I_1, I_2, I_3 \ldots., I_n\},\ I_n \in R^{W,H}$ of length N is divided into two parts, $\{1, (n-1)\}$ frame as input; nth as the label; The first $n\text{-}1$ frames are used as the input in the training process to extract features set $f_{I_{n-1}} \in R^{W,H,C}$; C is the number of channel; and then read the memory $Mem \in R^{M,C}$ to get the similarity index matrix $V \in R^{M,W*H}$. Then, update the memory module by V, and aggregate feature $f_{I_{n-1}}$ and Mem to obtain $Agg_f \in R^{2C,W*H}$. Following that, model restores the features $Agg_f \in R^{2C,W*H}$ to get the predicted $\hat{I}_{nth}$ frame. Finally, calculate the loss between the predicted $\hat{I}_{nth}$ frame and the real I_{nth} frame after retrieving the predicted value from the model. There are also some other additional loss functions applied during the training phase. In the following sections, we explain each module presented in Figure 1 that are used in our COVAD framework.

3.1 Encoders and Decoders

U-Net was originally designed as a CNN for image segmentation and has achieved excellent results in many international competitions [25,26]. Its unique structure and design philosophy inspired researchers in the field of computer vision, such as symmetrical ideas, up-sampling, and skip connections. The necessary functions of the CNN for video anomaly detection are to extract video feature frames and restore the feature to video frames through encoding/decoding process. The U-Net has a natural advantage that other network structures do not have, which is the symmetric structure of the network as shown in Figure 2. It consists of repeated applications of convolutions each followed by pooling at the extract feature phase and upsampling at the restore phase. For the upsampling, the max pooling is non-inevitable and therefore, it is possible to add switch variables recording the information of max pooling, such as the position of the maximum value. In the decode, the upsampling uses these switches to reconstruct current layer above into appropriate locations of next layer, preserving the structure of the stimulus [27].

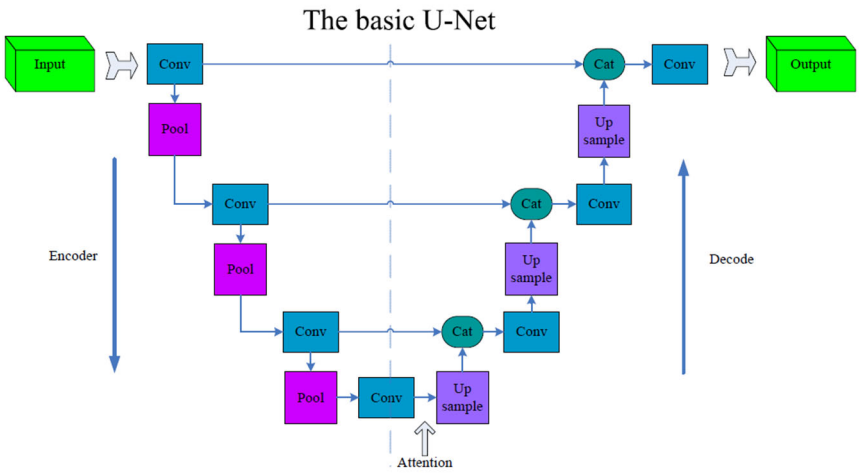


Figure 2 The basic U-Net: The U-Net network is composed of convolution, pooling, upsampling, and skip connections, where convolution and pooling are used to extract input features, upsampling is to restore the pooled and scaled features, and skip connections are feature splicing, trying to use a wider range of information to help restore video frames

At present, U-Net is widely used in video frame reconstruction and future frame prediction tasks. In addition, due to the skip connection of the U-Net network, fine-grained details can be recovered during prediction by extracting more video information during the decoding process. However, in the U-Net network structure, skip connections are not always useful specially for reconstruction tasks. This is mainly due to having noisy data in the previous feature set and not conducive to restore the most realistic features. Thus, skip connections are unrealistic to apply in this scenario. For the prediction-based video anomaly detection task in this paper focuses on the previous features that contains part of the information lost during the training process, and connecting the previous features to the current features can improve the accuracy of prediction [28]. The Attention in Figure 2 provides the interaction between video features and memory module.

Another issue with the strong representation ability of CNN is that the inability of defining the exact boundary between the normal event and abnormal event [14]. The final feature extracted from the encoding of the training phase, which obtained from normal data might deviate from the normal pattern, or out of its boundaries. In the testing phase, the features extracted from abnormal data may be regarded as normal features, resulting misclassification. Therefore, identifying and limiting the representation ability of neural network model is one of the most important aspects of network structure optimization. We removed the last batch of normalization [29] and ReLU layers [30] in the encoder, limiting different feature representations. We instead add an L2 normalization layer to make the features have a common scale.

2.2 Memory module

This module is composed of a randomly generated sparse matrix $M \times C$. The length and width of the matrix is M, depending on the actual application scenario, usually representing the number of normal behaviors in the training set, or the number of videos in the training set, or the number of different camera positions. The length of the feature extracted by the CNN is C, which is the same as the width of Memory. Here, the operation of reading and updating the memory module in this paper basically follows the processing in [14][22]:

Read: The read operation is to calculate the similarity between the query point and all the entries in the memory module, and find the closest entry and the second entry from the query point. The former is used to fit the query-worthy behavior pattern, and the latter is used to expand the class spacing, where there are two components of the loss function. Second, in the update operation, the weighted average of the query points is accumulated to the nearest entry by the L2 norm.

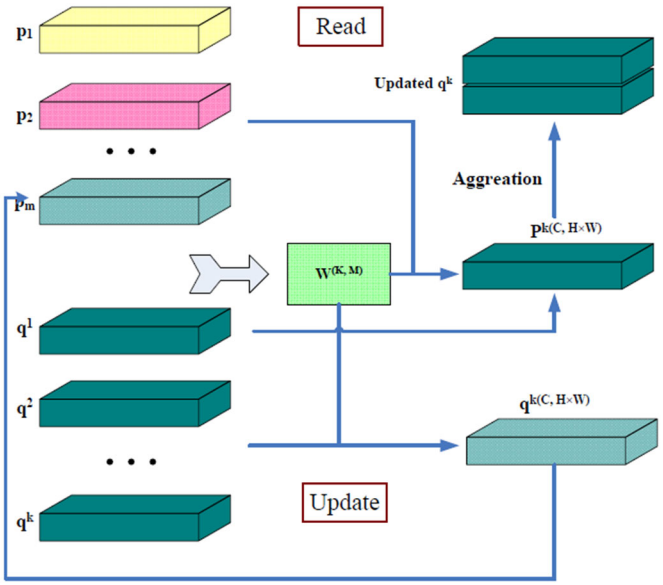


Figure 3 The algorithm flow of the memory module, including the flow chart of reading and updating memory

In the process of reading the memory module, first calculate the similarity between the query feature value and all the entries in the memory module, that is, the cosine similarity, But which is calculate by equation 3:

$$w^{k,m} = \frac{\exp\left(p_m^T q^k\right)}{\sum_m \exp\left(p_m^T q^k\right)} \qquad (1)$$

where p_m represents the entry in memory and q^k represents the query point, the encoded feature of the input video. So, we compute the similarity $w^{k,m}$ of the query point q^k to the memory module p_m as the weight of the memory module and read the memory module according to this weight $w^{k,m}$.

$$p^k = \sum_{m'=0}^{M} w^{k,m'} p_{m'} \qquad (2)$$

This paper reads all memory entries instead of the closest entry, to consider the integrity of the normal pattern, which is beneficial to get a more accurate model. because anomaly detection is essentially a binary classification problem. In the real scene, different normal patterns may coexist at the same time, and there is an interdependence between the normal.

Update We use the probabilities in equation 1 to select all the nearest query points corresponding to each memory. U^m is defined as the index set of the mth memory entry corresponding to the nearest query point, then the update mechanism is completed by the following equation.

$$p^m = \varphi\left(p^m + \sum_{k \in U^m} \hat{v}^{k,m} q^t\right) \qquad (3)$$

The weighted average is used here instead of \$sum\$, so that the query points closer to mth have a greater impact on the update of mth. The way of calculating $v^{k,m}$ is similar to equation 1, but the normalization is performed in the horizontal direction. Thus, $v^{k,m}$ can be expressed as in equation 4. After obtaining $v^{k,m}$, it should be normalized again following equation 5.

$$v^{k,m} = \frac{\exp\left((p_m)^T q^k\right)}{\sum_{k'=1}^{K} \exp\left((p_m)^T q^k\right)} \qquad (4)$$

$$\hat{v}^{k,m} = \frac{v^{k,m}}{\max\limits_{k \in U^m} v^{k,m}} \qquad (5)$$

Since the initial memory modules are randomly generated, there is no guaranteed that the distance between memory entries is sufficient. Therefore, the analyses in this paper incorporate a limit to the initial value of the randomly generated memory module R (Equation 6) to ensure that each entry is sufficiently independent, where I is an identity matrix, and $||.||$ is the Frobenius norm of the matrix.

$$R = ||CC^T - I||_F \qquad (6)$$

This function is used to limit initially generated memory modules to ensure that there is enough distance between different memory entries to distinguish them and prevent confusion.

This paper proposes another explanation scheme for the above memory module mechanism. In the process of multiple iterations, similar query points are continuously weighted and averaged to the nearest memory entry. In this paper, we propose a new memory module approach in which we assume that the memory entry corresponds to the clustering center of each normal event and its processing method is equivalent to k-means clustering. In the process of exploration, our analyses incorporate clustering loss to the iterative process of CNN, but did not achieve good results and therefore, in future we will explore how to reduce the loss.

3.3 Coordination Attention

The attention mechanism emerged as an improvement over the encoder decoder-based neural machine translation systems. Since video processing applications have no limitation on the length of the input and output sequences and need to allocate more computing resources, encoder decoder-based attention mechanisms are widely used [31,32].

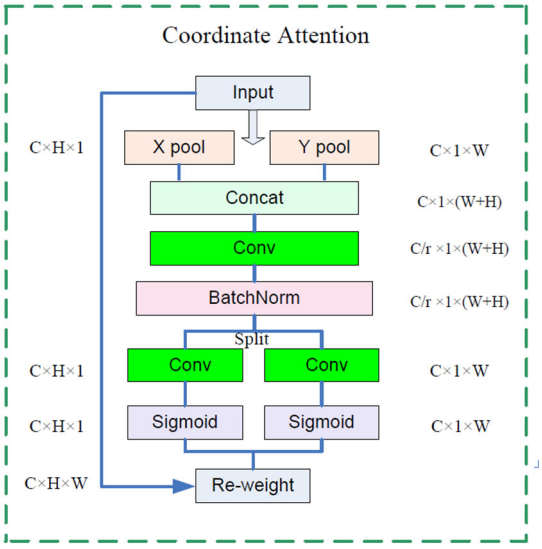


Figure 4 Coordinate Attention C is the number of channels; H, W represent the length and width of the current feature, respectively

Traditional channel attention allows neural networks to learn what should be focused on during the learning by allowing the network to iteratively focus on the attention of its filters. These channel attentions generally transform the feature tensor into a single feature vector through 2D global pooling. General attention-based algorithms often use attention pooling to encode global spatial information, but compressing the spatial information into one channel interpreter loses many features and it is difficult to preserve the spatial information. As it is important to preserve video features during the long-term interactions, it is required to improve the accuracy of visual tasks. In addition, the attention

module needs to acquire more precise spatial information, which help to capture the target of long-term interactions.

As channel attention mechanisms neglect positional information that helps to generate spatial information, we can embed coordinated attention mechanism to aggregate features along the spatial directions [24]. The coordinated attention mechanism consists of two steps: coordinate information embedding and coordinate attention generation. Figure 4 depicts the coordinate attention block that will be used to integrate with two steps encode channel correlations and long-term dependencies using precise location information.

Coordinate information embedding: Channel attention is established as two 1D feature encoding that aggregate these features along with two spatial directions. Therefore, long-term dependent features can be captured along one spatial direction, and precise location information is preserved along with the other. Given an input X, use two pooling kernels $(H, 1)$ and $(1, W)$ to encode all channels along with the horizontal and vertical directions. The cth channel information in the horizontal and vertical directions can be expressed as shown in Equation 7 and Equation 8.

$$z_c^h(h) = \frac{1}{W} \sum_{0<i<W} x_c(h, i) \qquad (7)$$

$$z_c^w(w) = \frac{1}{H} \sum_{0<i<H} x_c(j, w) \qquad (8)$$

Coordinated attention generation:

Coordinated attention generation follows three important steps for computer vision tasks:

- When designing the network structure, it should be designed as simple as possible and need to make sure that it does not utilize additional memory;
- The network should be able to understand the relationship between different channels, which is the key to the attention mechanism;
- According to the analysis and findings in this paper, the network should have the ability to capture the region of interest (the most important region) in the video with precise location information.

Once the coordinated attention has generated features of the embedded video, the connection information is sent to a shared convolutional transformation function F_1 in Equation 9, where $[.]$ represents a connection operation along the third spatial dimension.

$$f = \gamma\left(F_1([z^h, z^w])\right) \qquad (9)$$

The operation of splicing the weights in the w direction and the weights in the h direction into a weight matrix. The third spatial dimension generally refers to the dimension occupied by the channel. γ is a activate function. $f \in R^{\frac{C}{r}\times(H+w)}$, r is used to control the block size reduction ratio.

Then, we divide f into $f^h \in R^{\frac{C}{r}\times H}$, $f^w \in R^{\frac{C}{r}\times W}$. There are additional convolutional transforms F_h and F_w that transform $f^h$and f^w respectively into tensors with the same number of channels as the input X, yielding.

3.4 Model Evaluation Criteria

This section explains the loss function of the model proposed in this paper, the evaluation algorithm for detection accuracy and introduce the proposed visual explanation scheme to advance the detection accuracy from the frame level to the pixel level.

3.4.1 Loss function

In this work we use Following three main loss functions:

- the prediction error δ_{pred}
- the L2 norm loss between the query point and its nearest memory entry δ_{fit}, and
- the segmentation loss between the query point and the next closest memory entry δ_{sp}

We can also use the similarity loss for randomly generated memory entries as shown Equation \ref{R}, but this is not the training loss as the training loss consists of δ_{pred} , δ_{fit} and δ_{sp} and evaluated based on the Equation 10. However, we are not using the similarity loss to evaluate our model performances in this work.

$$\delta_{Train} = \delta_{pred} + \lambda_f \delta_{fit} + \lambda_s \delta_{sp} \qquad (10)$$

In prediction loss δ_{pred}, we minimize the L2 distance between the future frames $\hat{I}$ generated by the decoder and the true future frames I as shown in Equation 11.

$$\delta_{pred} = \sum_{w}^{W}\sum_{h}^{H}\left|\left|\hat{I}^{w\times H} - I^{w\times H}\right|\right|_2 \qquad (11)$$

The feature fit loss δ_{fit} encourages queries to be closer to the nearest item in the memory, which is computed by the L2 norm between them. Following Equation 12 shows the feature fit loss δ_{fit}, where $p_{q_t^k}$ is the memory entry closest to the query point q_t^k. This loss can also be considered as the clustering error.

$$\delta_{fit} = \sum_{k=1}^{K}\sum_{t=1}^{T}\left|\left|q_t^k - p_{q_t^k}\right|\right|_2 \qquad (12)$$

To ensure that different memory entries still maintain a certain distance during the updating and training process, we introduce the term δ_{sp} to prevent different memory entries from being confused during training by penalizing the distance between the query feature and the next closest memory entry.

$$\delta_{sp} = -\sum_{k=1}^{K}\sum_{t=1}^{T}\left\|q_t^k - p_{se}\right\|_2 \quad (13)$$

Loss functions used in the work are mentioned in Equation 11, 12 and 13, and results obtained from those functions considered together during the training phase to evaluate the performance of the model. The prediction loss is the most important loss function as the other two loss functions play a relatively small role and can be considered as secondary training loss functions. The λ_f and λ_s values used in Equation 10 usually ranges in between 0.1 and 0.01. The best fit values to our proposed models will be explored using different experiments and explain in the Section 4.

3.4.2 Model Evaluation

Video anomaly detection is a classification problem, and the ROC curve (Receiver Operating Characteristic curve) is one of the commonly used classification evaluation metrics [33]. The ROC curve plots two parameters: true positive rate (TPR) and false-positive rate (FPR). True positive rate is also called as Recall, which indicates the probability of an actual abnormal event will predict as a abnormal event. False positive rate indicates the probability of a true normal event will predicts as an abnormal event [34]. TPR and FPR can be expressed as mentioned in Equation 14 and Equation 15, respectively.

TP is the outcome when the model correctly predicts true abnormal event; FN is the outcome when the model incorrectly predicts true normal event and detected as an abnormal event; FP is an outcome when the model incorrectly predicts true normal and detected as an abnormal event, and TN is the model outcome when the model predicts true normal.

$$TPR(Recall) = \frac{TP}{TP + FN} \quad (14)$$

$$FPR = \frac{FP}{FP + TN} \quad (15)$$

The area under the ROC Curve (AUC) [35,36] provides an aggregate measure of performance across all possible classification thresholds. The Area Under the PR curve (AUC-PR) is useful when true negatives are more common than true positives. The PR curve only focuses on the predictions of the positive (rare) class and therefore, this is a good metric for anomaly detection. The difficulty lies in predicting those rare truly positive events. Accuracy is directly affected by the category or class imbalance effect and as a result, FP outcome of the model is also affected. Hence, the ROC curve does not capture this effect. For highly imbalanced datasets, the PR curves are more capable of highlighting differences between model outcomes. Therefore, for a highly unbalanced class setting, the AUC-PR score can be considered as the best metric to compare different models.

4 Experimental Results

This section explains different analyses conducted on video anomaly detection using above-mentioned datasets and selected hyperparameters.

4.1 The proposed COVAD approach

The following step-by-step process provides detailed information on how our proposed anomaly detection approach, called COVAD, is implemented and evaluated.

1) As the first step, our algorithm randomly generates memory modules, and build $M \in R^{m \times c}$ matrix according to the number of videos and behavioral patterns, where R represents the Equation 6, $\boldsymbol{m}$ is the number of normal behavior patterns and $\boldsymbol{c}$ represents the number of features per channel. Initially, the value of $\boldsymbol{m}$ is set to 10.
2) Next, read the dataset and divide it into multiple consecutive T frames. The first $t-1$ frame assigns as the input to the encoder network, and use convolution and downsampling to scale up the extracted features to make a $32 * 32 * 512$ feature space.
3) Following that, input the extracted features into the collaborative attention mechanism and re-allocate weights to obtain new video features.
4) Randomly generated memory module calculates its similarity, and updates the memory module according to the method explained in Section 3.2. It also aggregates the memory features and query features as the hyperparameters of the loss function. We conducted a set of experiments to identify the most suitable hyperparameter values as explained in Section 4.3.
5) Input the obtained aggregated features into the decoder network to restore the video.
6) Next, calculate the error between the restored video frame $\hat{I}_t$ and the real I_t frame.
7) Use backpropagation to update the network parameters until it minimizes the error.
8) Finally, classify the given input once the model converged to the minimum error point.

4.2 Dataset Description

The analyses in this work are mainly based on two different datasets: UCSD [15] and Avenue [16]. The UCSD dataset is a campus pedestrian dataset released by the University of California, San Diego in 2013, which contains two subsets called Ped1 and Ped2. The number of training videos sets used in Ped1 and Ped2 are 34 videos and 16 videos, respectively and this training set contains only normal frames. The test set contained both normal frames and exception frames and has 36 videos in Ped1 and 12 videos in Ped2. Frame-level annotations are provided for all test video clips and 10 of which have pixel-level ground truth. In this research, our analyses are mainly based on Ped2 as Ped1 was not pre-processed and it is a unlabeled dataset. The Avenue is a dataset released by the Chinese University of Hong Kong in 2013, which contains 15 videos of 2 minutes each. The total number of frames is 35240 and 8478 frames from 4 videos can be used as the training set. These videos contain typical unusual events including running and throwing objects.

4.3 Hyperparameter selection process

We conducted several experiments to select the best values for the hyperparameter λ_f and λ_s that is used in Equation 13 for calculating the total loss. To verify the effectiveness of these hyperparameters with different values in our analysis, we used UCSD-Ped2 dataset to verify the anomaly detection accuracy when λ_s and λ_f parameters assign 0.02, 0.04, 0.06, 0.08, and 0.1 values separately for different iterations. Accuracy of the COVAD model for different experiments are shown in Table 1. The accuracy does not show obvious regularity, but λ_f has a great influence on the detection results.

When the value of λ_f is 0.1, the experimental effect is relatively stable, and the detection accuracy is basically the highest value. Hence, in our analyses we set λ_f=0.1.

Table 1: The accuracy of anomaly detection under different value of hyperparameters λ_f, $\lambda_s$of Ped2

	λ_s					
λ_f	Value	0.02	0.04	0.06	0.08	0.1
	0.02	89.9	95.1	94.6	92.6	94.2
	0.04	92.5	95.3	94.9	90.4	88.2
	0.06	90.9	95.4	91.2	94.5	89.0
	0.08	93.1	83.2	84.6	91.2	89.3
	0.1	96.8	93.7	92.8	95.4	96.2

Table 1 shows the detection results under different hyperparameter values for λ_f and λ_s. The detection accuracy does not show a clear Gaussian distribution after fixing the value of one hyperparameter. The main reason behind results can be the relationship between three different nonlinear loss functions in Equation 13 or the insufficient training data.

Based on the results in Table 1, the two highest accuracy(average) are 96.8 and 96.2 are obtained when λ_f=0.1 and λ_s =0.1 or 0.02. Hence, we set both these hyperparameter values to 0.1 and previous works also proven that these hyperparameters exhibited better performances [14].

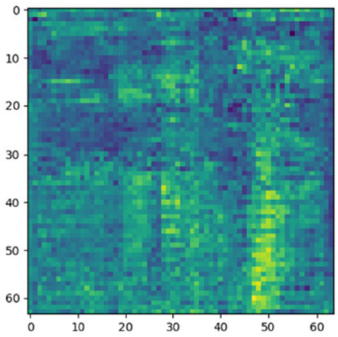
4.4 Effectiveness of the attention mechanism in video anomaly detection

In this section, we analyze the outcome of the COVAD model to explore whether the attention mechanism improves the accuracy of video anomaly detection. To verify this, we randomly selected an image (Figure 5(a)) from the Avenue dataset and extracted its features before and after adding coordinated attention using our COVAD model and MNAD [14] model. The comparison process has been done using the following steps.

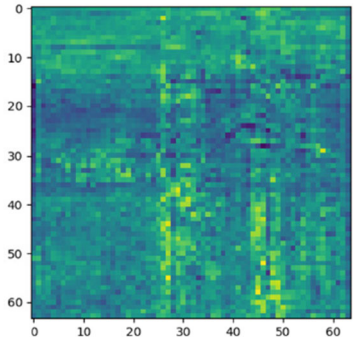
1. MNAD and COVAD networks are trained separately. The MNAD network is implemented without using the attention mechanism, and the COVAD is implemented using coordinated attention.
2. Selected a video frame randomly and then, input it into the above-mentioned two trained networks.

3. Generated the feature map for the encoder and observed difference between outputs of two networks.

a



b



c

Figure 5. The a is *a* frame in the video, *b* is the feature map generated without a coordinated attention mechanism, and *c* is the feature map generated by the coordinated attention mechanism.

Figure 5 depicts how the weight redistribution of the coordinated self-attention mechanism helps the neural network to focus on meaningful targets having the effect of anti-noise, and how it helps to improve the detection efficiency. Figure 5(b) clearly indicates that without the coordinated attention mechanism the upper part of the video frame is relatively dark. As a result, after reading this video frame, the RGB value of this area gets relatively large and hence, this will affect on the neural network training and model performances. Since this dark area (in this research we refer this as dark area as background) is not important in the classification, training the neural network model using this types of frames are the best practice and it consume lots of resources when the model gets larger. Once we apply the trained coordinated attention, we can clearly observe that the object distribution in Figure 5(a) is more visible on the feature map shown Figure 5(c) compared with the feature map shown in Figure 5(b) based on its contrast and dark colors. This indicates that network parameters used in our paper are more reasonable and help to obtain more effective features and more realistic video frames.

4.5 Testing environment and model performances

The testing environment is Tesla V100 Volta P100 GPU Accelerator with a 32GB Graphics Card and few models are executed at the Google Colab. Since we tested a large number of hyperparameters, the part of the validation experiments was run in Colab.

Compared with previous networks, proposed network in this paper has good time efficiency when running tests. During the training phase, it costs 14 hours for Avenu dataset and 8 hours for Ped2 dataset on V100. During the testing, the COVAD model process 28 frames per second under P100.

Table 2 Quantitative comparison of the frame-level AUC-PR results of our COVAD method with the state-of-the-art models.

Method	UCSD(Ped2)	Avenue	Techniques
AMDN [37]	90.8%		DFF+SVM
Unmasking [38]	82.2%	80.6	Unmask
StackRNN [9]	92.2%	81.7	TSC+sRNN
MemAE [22]	91.7%	81.0	Memory module
MNAD [14]	94.2%	80.6%	U-net
COVAD	**96.5%**	83.4%	CA

Table 2 shows the quantitative comparison results of our COVAD method and the state-of-the-art methods on frame-level AUC-PR results. Previous research on detecting anomalies in video frames [39,40] focus on the content of the abnormal and the variant of represent learning. Admittedly, these papers have achieved good results, but no breakthrough in the proposed core technology

Based on the results in Table 2, the COVAD method can effectively improve the accuracy of anomaly detection compared to other baseline models. We can find that the COVAD method obtained the highest AUC value for both UCSD-Ped2 and Avenue datasets.

Another most important finding in this work is on the reduction model convergence time, which is mainly due to the integration of the attention mechanisms in our COVAD approach. Since the self-attention mechanism is lightweight and mobile-level, it does not take a lot of time for training and testing. Therefore, compared with previous methods, our method has less convergence time.

4.6 Model comparison using ROC Curve

Based on the results in Table 2, the highest performance of the models exhibited by the UCSD-Ped2 dataset. Therefore, in order to analyse the detailed measure of the classifiers we generated the ROC curve as depicted in Figure 6 for COVAD and MNAD algorithms.

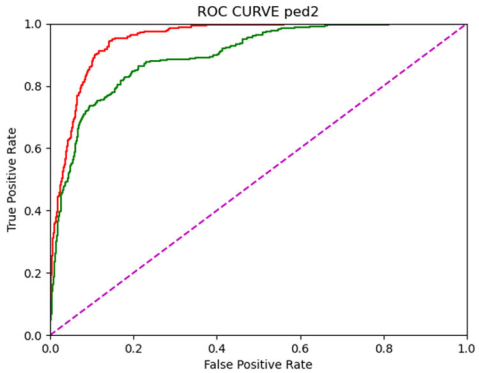


Figure 6 The ROC Curve of Ped2 under the COVAD and MNAD; Red color represents the COVAD model and green color represents the MNAD model

In Figure 6, the horizontal axis represents false positive rate, the vertical axis represents true positive rate, and the purple dotted line represents random cases in which the probability of the result being false positive and true positive is 50% each. The red solid line represents the COVAD algorithm under different thresholds and the prediction accuracy of the green solid line represents the MNAD algorithm. The area under the solid line represents the AUC and the higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes. Based on the results in Table 2, the highest performance of the models exhibited by the UCSD-Ped2 dataset. In order to analyze the detailed measure of the classifiers we generated the ROC curve as depicted in Figure 6 for COVAD and MNAD algorithms. From the Figure 6, it can be concluded that the MNAD classifier is able to detect more number of positives and True negatives than False negatives and False positives. Hence, the prediction result of the COVAD algorithm is better than that of the MNAD algorithm.

4.7 A Visual Test

To build a visual exception explanation, we propose a visual anomaly test to detect video anomalies in a sample video. First, the error between the predicted video frame and the real video frame is calculated to obtain the error feature map, and then the real video frame is detected by the target detector to obtain each target frame. Next, calculate the average error within each target box and through multiple tests, a reasonable threshold is set that determines which target boxes are abnormal areas and should be marked in red.

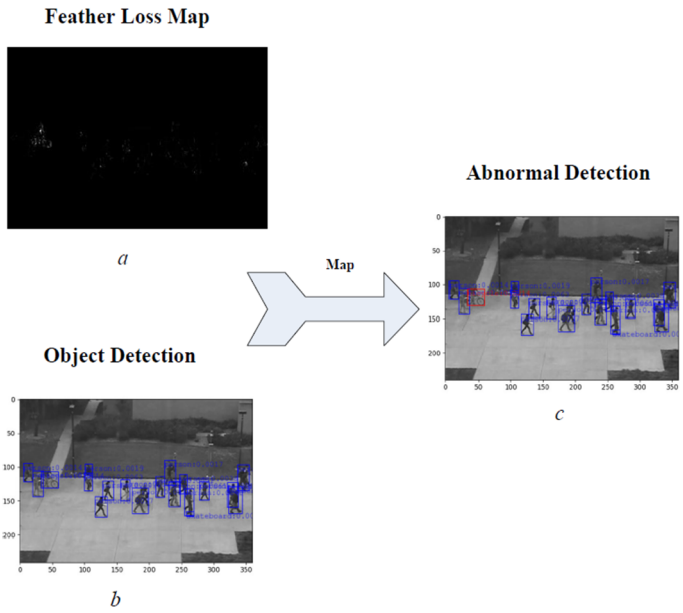


Figure 7 The visual testing. *a* is the feather loss map which computed by the pixel error between predicted frame and real frame; b is result from the object detection; we obtain c by matching a and b.

Figure 7 is our experimental result. In the specific implementation process, the feature error map is calculated by the feature subtraction of the restored video frame and the directly read real video frame. The object detector is implemented by Retina-net single-stage object detector, and the network used is resnet50, and the performance is sufficient. The white bright spots in Figure 7.a represent areas with large errors, black represent areas with small errors. The object detector is only responsible for object detection, we detect the truth future frame by Retina-net single-stage object detector, and 7.b is the result of the object detector. In 7.b, each blue box represents a target, which is usually the subject where the anomaly occurred. Abnormal judgment is obtained by calculating the object frame mean error of future frame. The abnormal area is obtained by matching the target boxes of both 7.a and 7.b, as shown 7.c. Therefore, we can conclude that the pixel error of the abnormal occurrence area is larger than that of other areas. Thanks to this process, even unsupervised learning models can provide a visual interpretation of video anomaly detection.

6 Conclusion and Future work

With the improvement of computer hardware and network bandwidth, video will definitely become the main medium for transmitting information in the future and this is one main reason to attract many researchers towards computer vision. In this paper our main focus is to detect anomalies in surveillance videos that are deployed in different locations, such as highways, schools, prisons, etc. The manual inspection of video anomaly detection in real time is not very efficient due to the discontinuity of human eye monitoring over the time. The algorithm proposed in this paper incorporates a coordinated self-attention mechanism to help the neural network to focus on meaningful objects during training by ignoring the background in the video. Based on the experimental results, our proposed algorithm can avoid the detection efficiency of unimportant background noise, that is, the algorithm in this paper has a strong anti-

noise ability. Many unsupervised video anomaly detection approaches proposed in the literature have used frame-level objective function as the training loss function, and then detect the abnormal area through the splicing Object detection algorithm. This approach seems to achieve pixel-level video anomaly detection, but this is difficult to achieve in the actual deployment process. Compared with video anomaly detection, the network structure of video Object detection is more complex, and it is difficult to establish a joint algorithm framework to connect the two neural networks. Therefore, the best solution is to establish an anomaly detection mechanism centered on Object and Behaviors and therefore, our objective it to fill this gap in the literature. We proposed COVAD, a model to detect video anomalies and achieved better performance compared to many baseline models.

The direct detection of abnormal regions in the real-time video is one of our ultimate goals related to this reach. In future, we aim to implement an unsupervised video anomaly detection network that can be jointly trained with the pixel-level object detection network. The purpose of detecting video anomalies is to solve the issues that occur in real-time, that is, to eliminate disasters that have not yet occurred. Therefore, the response mechanism in the actual deployment stage is also worthy of our consideration in future.

.

Declaration of Competing Interest We declare that we have no conflict of interest.

References

1 O. P. Popoola and K. Wang, "Video-Based Abnormal Human Behavior Recognition—A Review," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 6, pp. 865-878, Nov. 2012 DOI: 10.1109/TSMCC.2011.2178594.

2 B. Ramachandra, M. J. Jones and R. R. Vatsavai, "A Survey of Single-Scene Video Anomaly Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 5, pp. 2293-2312, 1 May 2022, DOI: 10.1109/TPAMI.2020.3040591.

3 Suarez, Jessie James P., and Prospero C. Naval Jr. "A survey on deep learning techniques for video anomaly detection." arXiv preprint arXiv:2009.14146 (2020).

4 C. Yu, X. Zheng, Y. Zhao, G. Liu and N. Li, "Review of intelligent video surveillance technology research," Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology, 2011, pp. 230-233 DOI: 10.1109/EMEIT.2011.6022904

5 Zhu, Sijie, Chen Chen, and Waqas Sultani. "Video anomaly detection for smart surveillance." arXiv preprint arXiv:2004.00222 (2020).

6 Kiran, B. Ravi, Dilip Mathew Thomas, and Ranjith Parakkal. "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos." Journal of Imaging 4.2 (2018): 36. DOI: 10.3390/jimaging4020036

7 Hasan, Mahmudul, et al. "Learning temporal regularity in video sequences." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016 DOI: 10.1109/CVPR.2016.86

8 W. Luo, W. Liu and S. Gao, "Remembering history with convolutional LSTM for anomaly detection," 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 439-444 DOI: 10.1109/ICME.2017.8019325

9 W. Luo, W. Liu and S. Gao, "A Revisit of Sparse Coding Based Anomaly Detection in Stacked RNN Framework," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 341-349
DOI: 10.1109/ICCV.2017.45

10 W. Liu, W. Luo, D. Lian, and S. Gao, “Future Frame Prediction for Anomaly Detection - A New Baseline,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, Jun. 2018, pp. 6536–6545.
DOI: 10.1109/CVPR.2018.00684

11 Ye, Muchao, et al. "Anopcn: Video anomaly detection via deep predictive coding network." Proceedings of the 27th ACM International Conference on Multimedia. 2019.
DOI: 10.1145/3343031.3350899

12 Y. Lu, K. M. Kumar, S. s. Nabavi and Y. Wang, "Future Frame Prediction Using Convolutional VRNN for Anomaly Detection," 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2019, pp. 1-8
DOI: 10.1109/AVSS.2019.8909850

13 W. Wang, F. Chang, and C. Liu, “Mutuality-oriented reconstruction and prediction hybrid network for video anomaly detection,” SIViP, Jan. 2022,
DOI: 10.1007/s11760-021-02131-w

14 H. Park, J. Noh, and B. Ham, “Learning Memory-Guided Normality for Anomaly Detection,” in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, Jun. 2020, pp. 14360–14369
DOI: 10.1109/CVPR42600.2020.01438

15 Li W, Mahadevan V, Vasconcelos N. Anomaly detection and localization in crowded scenes[J]. IEEE transactions on pattern analysis and machine intelligence, 2013, 36(1): 18-32
DOI: 10.1109/TPAMI.2013.111

16 Lu C, Shi J, Jia J. Abnormal event detection at 150 fps in matlab[C]//Proceedings of the IEEE international conference on computer vision. 2013: 2720-2727
DOI: 10.1109/ICCV.2013.338

17 V. Saligrama and Zhu Chen, “Video anomaly detection based on local statistical aggregates,” in 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, Jun. 2012, pp. 2112–2119
DOI: 10.1109/CVPR.2012.6247917

18 K. Cheng, Y. Chen and W. Fang, "Video anomaly detection and localization using hierarchical feature representation and Gaussian process regression," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 2909-2917
DOI: 10.1109/CVPR.2015.7298909

19 Zhang, Ying, et al. "Video anomaly detection based on locality sensitive hashing filters." Pattern Recognition 59 (2016): 302-311.
DOI: 10.1016/j.patcog.2015.11.018

20 M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, “Learning Temporal Regularity in Video Sequences,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 733–742
DOI: 10.1109/CVPR.2016.86

21 W. Luo, W. Liu, and S. Gao, “Remembering history with convolutional LSTM for anomaly detection,” in 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, Hong Kong, Jul. 2017, pp. 439–444
DOI: 10.1109/ICME.2017.8019325

22 D. Gong et al., “Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection,” in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), Oct. 2019, pp. 1705–1714

DOI: 10.1109/ICCV.2019.00179

23 M.-I. Georgescu, A. Barbalau, R. T. Ionescu, F. Shahbaz Khan, M. Popescu, and M. Shah, “Anomaly Detection in Video via Self-Supervised and Multi-Task Learning,” in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, Jun. 2021, pp. 12737–12747
DOI: 10.1109/CVPR46437.2021.01255

24 Q. Hou, D. Zhou, and J. Feng, “Coordinate Attention for Efficient Mobile Network Design,” in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, Jun. 2021, pp. 13708–13717
DOI: 10.1109/CVPR46437.2021.01350

25 Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.
DOI: 10.1007/978-3-319-24574-4_28

26 Baheti, Bhakti, et al. "Eff-unet: A novel architecture for semantic segmentation in unstructured environment." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2020.

27 M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” in Computer Vision–ECCV 2014, vol. 8689, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833

28 Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh and J. Liang, "UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation," in IEEE Transactions on Medical Imaging, vol. 39, no. 6, pp. 1856-1867, June 2020
DOI: 10.1109/TMI.2019.2959609

29 Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." International conference on machine learning. PMLR, 2015.

30 Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).
DOI: NIPS2012_c399862d

31 Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu. "A review on the attention mechanism of deep learning." Neurocomputing 452 (2021): 48-62
DOI: 10.1016/j.neucom.2021.03.091

32 Shaw P, Uszkoreit J, Vaswani A. Self-attention with relative position representations[J]. arXiv preprint arXiv:1803.02155, 2018.

33 McClish, Donna Katzman. "Analyzing a portion of the ROC curve." Medical decision making 9.3 (1989): 190-195.
DOI: 10.1177/0272989X8900900307

34 Bradley, Andrew P. "The use of the area under the ROC curve in the evaluation of machine learning algorithms." Pattern recognition 30.7 (1997): 1145-1159.
DOI: 10.1016/S0031-3203(96)00142-2

35 Janssens, A. Cecile JW, and Forike K. Martens. "Reflection on modern methods: revisiting the area under the ROC curve." International journal of epidemiology 49.4 (2020): 1397-1403.
DOI: 10.1093/ije/dyz274

36 Fan, Zi-Chen, et al. "AUC optimization for deep learning-based voice activity detection." ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019.
DOI: 10.1109/ICASSP.2019.8682803

37 Xu, Dan, et al. "Detecting anomalous events in videos by learning deep representations of appearance and motion." Computer Vision and Image Understanding 156 (2017): 117-127.
DOI: 10.1016/j.cviu.2016.10.010

38 Tudor Ionescu, Radu, et al. "Unmasking the abnormal events in video." Proceedings of the IEEE international conference on computer vision. 2017
DOI: CoRR abs/1705.08182 (2017)

39 Ganokratanaa, Thittaporn, Supavadee Aramvith, and Nicu Sebe. "Video anomaly detection using deep residual-spatiotemporal translation network." Pattern Recognition Letters 155 (2022): 143-150.
DOI: 10.1016/j.patrec.2021.11.001

40 Ganokratanaa, Thittaporn, and Supavadee Aramvith. "Generative adversarial network for video anomaly detection." Generative Adversarial Networks for Image-to-Image Translation. Academic Press, 2021. 377-420.
DOI: 10.1016/B978-0-12-823519-5.00011-7