Video Anomaly Detection using Convolutional Spatiotemporal Autoencoder

Rashmiranjan Nayak
Department of ECE
NIT Rourkela
Odisha-769008, India
rashmiranjan.et@gmail.com

Umesh Chandra Pati
Department of ECE
NIT Rourkela
Odisha-769008, India
ucpati@nitrkl.ac.in

Santos Kumar Das
Department of ECE
NIT Rourkela
Odisha-769008, India
dassk@nitrkl.ac.in

***Abstract*—A convolutional spatiotemporal autoencoder is used for video anomaly detection. The proposed model architecture comprises of three major sections, such as spatial encoder, temporal encoder-decoder, and spatial decoder. The spatial encoder is implemented using three layers of the convolutional layers. Then, the temporal encoder-decoder is realized with the help of Convolutional Long Short Term Memory (ConvLSTM), gated with the tanh and sigmoid activation functions. Finally, the spatial decoder is implemented using three layers of deconvolutional layers. The proposed model is trained only on the dataset comprises the normal classes by minimizing the reconstruction error. Later, when the trained model is tested using the test dataset susceptible to contain anomalous activities, then high reconstruction error has resulted. Subsequently, a high anomaly score and low regularity score has resulted. When the regularity score of the frames falls below the set threshold level, then the corresponding frames are treated as anomalous ones. The proposed model is trained and tested on UCSD Ped1 and Ped2 dataset successfully. The results of the performance evaluation are found to be promising.**

***Index Terms*—Autoencoders, deep learning, convolutional LSTM, spatiotemporal models, video anomaly detection**

I. Introduction

Intelligent video surveillance systems can be widely used in various public places such as smart cities, markets, banks, shopping malls, streets, etc. to increase public safety by automatically detecting anomalous events such as crimes, road accidents, stampede, etc. Usually, the anomalous events are less frequent as compared to normal events and also very much specific to the application or context. Generically anomaly can be defined as an observation that significantly deviates from the other observations in the same context to evoke an intuition that is generated by a different mechanism [1]. In other words, outliers diverging from the trained model are regarded as anomalous events [2]. Manual monitoring of the video surveillance systems is a tedious, time-consuming, erroneous, and complicated task for the unworkable ratio of cameras to the human operators. Subsequently, there is a massive demand for efficient computer vision algorithms to detect video anomalies automatically. For practical applications, the video-based anomaly detection system should timely notify when activity deviates from the normal pattern and identify the time duration in which the anomaly has occurred [3]. Anomaly detection is an unsupervised learning technique that is used to identify the abnormal patterns or trends present in the data [4]. The video anomaly detection is different from supervised video analysis problems such as action recognition, event detection, etc. in two crucial challenges. Firstly, the video data is an unbalanced one between the positive and negative classes, i.e., generally, the positive examples (anomalous events) are fewer than the regular events. Secondly, the high variance within the positive classes as anomalous events may contain a large variety of different classes [2].

Basically, there are three types of modeling approaches for video anomaly detection, such as reconstruction models, predictive models, and generative models [4]. Here, the objective to reconstruct the frames of the video with minimum reconstruction error. In reconstruction modeling, various techniques such as Principal Component Analysis (PCA) and some varieties of the Autoencoders (AEs) are used for the effective representation of the features of the normal behaviors in the surveillance videos. The models are trained using only normal video sequences. During reconstruction, the abnormal or anomalous behaviors results in high reconstruction score. In the case of predictive modeling or Spatiotemporal modeling, both the spatial patterns and the temporal patterns of the video sequences are used for the pattern analysis. Here, the objective is to model the conditional distribution $P\left(X_t/(X_{t-1}, X_{t-2}, ..., X_{t-m})\right)$ and predict the current frame X_t or its encoded representation using the past frames $X_{t-1}, X_{t-2}, ..., X_{t-m}$, where m represents the number of past frames. The popularly known predictive models are autoregressive models and convolutional Long-Short-Term-Memory (LSTM) models. In the case of generative models, the objective is to model the likelihood of the normal video sequences in an end-to-end deep learning framework. The popularly known generative models are Variational autoencoders (VAE), Adversarially trained Auto-encoders (AAE), and Generative Adversarial Networks (GAN). Learning the temporal regularity using only regular videos during the training can be treated as an unsupervised task [5]. One of the state-of-art approaches for this type of modeling involves sparse coding and bag-of-words. However, in the case of bag-of-words prior information about the number of information is required, and also Spatiotemporal structure of the words is not preserved. Further, the optimization process involved in sparse coding is

computationally expensive for the videos.

Deep neural networks (DNNs) based approaches are found to be useful in event detection and recognition. However, they are found to be impractical for real-time video anomaly detection due to the sparsity of the positive data samples [2]. Tracking based approaches are suitable in anomaly detection in a simple and sparse environment, not in crowded or complex environments [6]. Hence, there is a demand for Spatiotemporal based deep learning models, which can detect the video anomaly in real-time with high accuracy and minimal time latency. Recently, a Spatiotemporal deep autoencoder is proposed for video anomaly detection [2]. However, it can detect only the anomaly happening, not the type of anomalous events. A DNN consists of a stack of convolutional auto-encoders is used to process the video frames to capture the spatial structures and grouped to extract the temporal features for the automatic video anomaly detection [6]. Various approaches such as 3D convolutional networks [7], robust deep autoencoders [8], deep convolutional auto-encoders [9], multiple instant learning [3], convolutional long short-term memory (LSTM) [10], Spatiotemporal auto-encoding for crowd anomaly [1], hybrid Spatio-temporal autoencoder [11] have been proposed for detecting various video anomalous activities. Though many research works have been attempted to improve the performance of the video anomaly detection, still, there are lots of gray areas to be improved concerning the online performances with competitive accuracy.

In this regard, an improved approach using convoloutional spatiotemporal autoencoder for the detection of video anomaly is proposed and implemented. Here, automatic feature extraction is performed using representation learning with the help of convolutional spatiotemporal autoencoder, regularity score is calculated using the reconstruction error, and anomaly detection based on the given threshold.

The rest of the paper is organized as follows. Section II describes the problem formulation. The methodology used for the video anomaly detection are discussed in Section III. The experimental results are discussed in Section IV, followed by conclusions in Section V.

II. Problem Formulation

Usually, the task of anomaly detection is treated as an unsupervised learning problem when there is no direct available information about the positive class, i.e., anomaly cases. However, practically most of the time, there is the availability of direct information about the negative class, i.e., normal classes or classes having no anomalies. Hence, in this case, anomaly detection can be treated as a semi-supervised learning problem [4]. The normal class distribution D_N can be estimated using the training samples comprise of only normal video sequences $x_i \in X_{train}$ by building an automatic representation $f_\theta : X_{train} \rightarrow R$ which minimizes the reconstruction error or reconstruction cost

$$\theta^* = \arg\min_{\theta} \sum_{x_i \in X_{train}} \|x_i - f_\theta(x_i)\|^2 \qquad (1)$$

over all the training samples, over all i. Further, the deviation of the test samples comprise of both positive and negative classes, i.e., $x_j \in X_{test}$ under the same representation is evaluated as the anomaly score. Video anomaly is said to be occurred when the anomaly score exceeds the set threshold.

In other words, for a given training frame sequences of a video, $X_{train} \in R^{N_{train} \times r \times c}$, which contains only normal classes and a test frame sequences of a video $X_{test} \in R^{N_{test} \times r \times c}$ which may contain both normal as well as anomalous classes, the video anomaly detection task is to associate each frame with an anomaly score corresponding to the saptiotemporal variations.

III. Methodology

The proposed video anomaly detection system is based on the intuition that the anomalous events will generate a high value of anomaly score (i.e., a low value of regularity score) as the trained model can't reconstruct the anomalous frames efficiently. A modified deep learning network inspired by the learning of temporal regularity in video sequences [5] and use of spatiotemporal autoencoder for abnormal event detection [6], is trained in an end-to-end fashion. Based on the regularity score, the anomaly can be detected corresponding to the set threshold. The methodology can be explained in three significant steps: preprocessing of data, representation learning for automatic feature extraction, and regularity score estimation form the reconstruction error.

A. Preprocessing

The raw frames of the input video data are converted to an acceptable resolution by the model. The pixel values are normalized to the range from 0 to 1 for ensuring that all the frames are in the same scale. The frames are also converted to grayscale to reduce the computational complexity by reducing the dimensionality [6]. The deep learning models are the data-driven approaches, and the availability of the proper dataset of the video anomaly detection is highly unlikely. Hence, data augmentation is used in the temporal dimension to increase the size of the training dataset [5]. This can be achieved by concatenating various strides of ten frames taken from the train video sequences in different order.

B. Representation Learning

The process of automatic feature learning to find proper representations of the input space which helps the predictors and classifiers to extract the useful information is known as representation learning [4], [12]. Here, a modified version of convolutional spatiotemporal autoencoder as mentioned in [6] is used to learn both spatial and temporal features of the input video sequence. Then, the regularity (regular motion patterns) [5] is measured with semi-supervised learning technique. Subsequently, this regularity score is used to detect the video anomaly. The representation learning is achieved with three major steps such developing model architecture, training with proper initialization and optimization, and calculating the regularity score.

1) Model Architecture: The proposed model architecture is an improved version of the similar works as reported in [5], [6] [13]. Autoencoders are the class of neural networks trained by back-propagation and used for representation of both linear as well as nonlinear transformations on the video sequences [4]. The proposed network as represented in Fig. 1 works in three steps: learning of spatial structures of each frame by spatial encoder, learning of temporal patterns of the encoded spatial structure by temporal encoder-decoder, and decoding the encoded spatial structures to reconstruct the image by spatial decoder. A temporal sliding window of size $T = 10$ is used for the temporal encoder-decoder. This based on the inference that more discriminative regularity score is resulted from the increasing value of T. However, increasing values of T makes the training process slower [5]. Hence, it is found that $T = 10$ provides an acceptable trade-off between the training time and the discriminative ability of the model. The network takes the input of video sequence of length T and tries to reconstruct input with minimum reconstruction error. The numbers mentioned in the output size are in the form of "$T \times$ Resolution of the frame $\times$ Number of filters". The spatial encoder processes one input frame at a time to encode the spatial structures. Once T number of frames have been processed, a feature vector is created by concatenating the encoded features of the T number of frames. Then, this feature vector is processed by the temporal encoder for encoding the temporal patterns such as motion. Finally, both the decoders, i.e., temporal and spatial decoders reconstruct the the video sequence with the help of inverse transformations. Here, two dimensional convolution and LSTM operations are preferred to make the network computationally efficient suitable for the practical scenarios. This based on the intuition that anomaly detection is a coarse level of understanding and further the anomalous video segments can be classified using more computationally expensive networks for better accuracy [3].

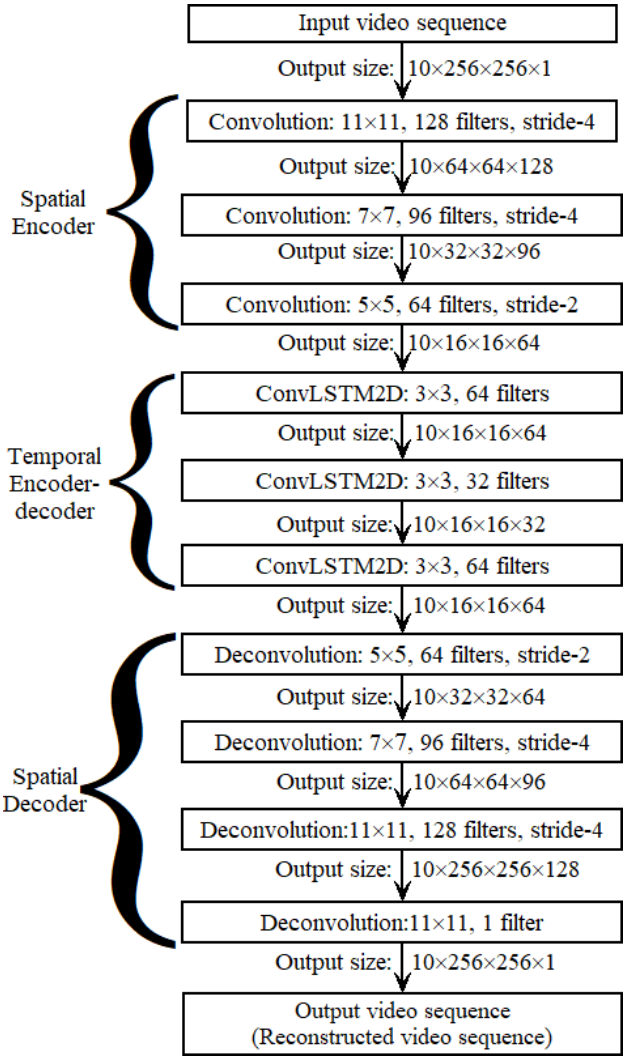

Fig. 1: Proposed network architecture.

The spatial encoder comprises of three convolutional layers and spatial decoders comprises of three deconvolutional layers as shown in the network architecture. Convolution operation is preferred for the spatial feature learning as it preservers the spatial relationships among the pixels of a frame [6]. Further, the temporal encoder-decoder is implemented using three-layer Convolutional Long Short Term Memory (ConvLSTM) model [14]. Matrix operations are replaced by convolutions in ConvLSTM as compared to the fully connected LSTM. ConvLSTM builds better spatial feature maps with fewer weights while applied to the input-to-hidden and hidden-to-hidden connections. The ConvLSTM models the spatio-temporal correlations using its convolotional layers [4]. The formulation of the ConvLSTM unit can be represented in Eq. 2 - Eq. 9 [6], [14]. Sigmoid activation function as expressed in Eq. 8 is used in the proposed model, specifically for the gating functions for the three gates (in, out, forget or recurrent) in the proposed ConvLSTM model as its value always is either 0 (no flow) or 1 (complete flow) of the inflammation throughout the gate.

The forget layer is represented in Eq. 2.

$$f_t = \sigma\left(W_f * [h_{t-1}, x_t, C_{t-1}] + b_f\right) \tag{2}$$

The new information addition can be expressed as Eq. 3 and 4.

$$i_t = \sigma\left(W_f * [h_{t-1}, x_t, C_{t-1}] + b_i\right) \tag{3}$$

$$\hat{C}_t = tanh\left(W_C * [h_{t-1}, x_t, x_{t-1}] + b_C\right) \tag{4}$$

The new and old information can be combined as Eq. 5.

$$C_t = f_t \otimes C_{t-1} + i_t \oplus \hat{C}_t \tag{5}$$

The outputs that has been learned so far to the convLSTM unit at the next step can be expressed in Eq. 6 and 7.

$$o_t = \sigma\left(W_o \otimes [h_{t-1}, x_t] + b_o\right) \tag{6}$$

$$h_t = o_t \otimes tanh\left(C_t\right) \tag{7}$$

The activation functions can be expressed as Eq. 8 and 9.

$$sigmoid\,(x) = \frac{1}{1+e^{-x}} \tag{8}$$

$$\tanh\,(x) = \frac{2}{1+e^{-2x}} - 1 \tag{9}$$

Here, x_t, h_t, C_t, W, b and $\otimes$ represent the input vector, the hidden state, cell state at time t, trainable weight matrices, bias vectors and Hadamard product respectively.

2) Initialization and Optimization: The proposed model is trained using ADAM (ADAptive Moment estimation) optimizer [15] with hyper parameters such as learning rate = 0.00001, first momentum decay = 0.9, and second momentum decay = 0.999. The Adam is a simple and computationally efficient algorithm for gradient-based optimization of stochastic objective functions and hence, suitable for machine learning problems with large data-sets and/or high-dimensional parameter spaces [15]. The initialization of weights are done using Xavier algorithm [16]. The scale of initialization based on the number of neurons present in input and output layers for reasonable signal strength throughout the multiple layers of the deep neural networks, is automatically determined by the Xavier algorithm [5]. Here, the objective is to train the model such that the reconstruction error $e_{reconst}\,(t)$ is minimized for a given normal class dataset.

3) Regularity Score: The reconstruction error for a given pixel with intensity I at a location (x, y) in a particular frame at time instant t can be calculated from the trained model using Eq. 10 [5].

$$e\,(x,y,t) = \|I\,(x,y,t) - f_W\,(I\,(x,y,t))\|_2 \tag{10}$$

Here, f_W is the learned model by the ConvLSTM autoencoder. Subsequently, reconstruction error of the particular frame at t with known pixel level reconstruction error can be calculated by using Eq. 11. Further, the anomaly score $S_{ano}\,(t)$ in the range of 0 to 1, can be calculated using Eq. 12 [6]. Finally, regularity score $S_{reg}\,(t)$ can be calculated using Eq. 13 [6].

$$e_{reconst}\,(t) = \sum\nolimits_{(x,y)} e\,(x,y,t) \tag{11}$$

$$S_{ano}\,(t) = \frac{e_{reconst}\,(t) - e_{reconst_{\min}}\,(t)}{e_{reconst_{\max}}\,(t)} \tag{12}$$

$$S_{reg}\,(t) = 1 - S_{ano}\,(t) \tag{13}$$

C. Video Anomaly Detection

The individual frames are checked whether a particular frame is anomalous (positive case) or normal (negative case) based on the associated anomaly score $S_{ano}\,(t)$ or regularity score $S_{reg}\,(t)$. When the value of $S_{ano}\,(t)$ corresponding to a frame at t exceeds the set threshold value θ_{th}, then the corresponding frames are treated as the anomalous frame. Conversely, when the value of $S_{reg}\,(t)$ goes below the set threshold value θ_{th}, then the corresponding frames are treated as the anomalous frame. The threshold θ_{th} is set tactically corresponding to the application and required sensitivity level. When the value of θ_{th} is very low, there is high possibility of getting false alarms and when the value of θ_{th} is very high there is a possibility of missed out the real anomaly. Hence, setting a proper threshold θ_{th} is always the requirement specific.

IV. Experimental Results and Analysis

A. Dataset

The proposed model is trained on one of the most used bench-marking datasets, i.e., UCSD Ped1 and UCSD Ped2 [17]. Here, all the training video sequences contain only normal activities and all the testing video sequences contain both normal as well as anomalous activities. The anomaly of these two datasets comprise of bikers, skaters, carts, wheelchairs, and people walking in the grass area. Each video clip in the Ped1 dataset contains 200 number of frames (fixed one) where as that of Ped2 contains variable number of frames.

B. Experimental Setup

The experiments for the implementation and validation of the proposed architecture are conducted using the experimental setup as shown in Table I.

TABLE I: Experimental Hardware and Software Setup

Hardware platform	
High-end system	Graphical computation system
CPU	Intel Xeon W-2123 (64-bit, 4 cores)
GPU	NVIDIA RTX 2080 Ti(11 GB)
RAM	DDR4 (64 GB)
Software platfrom	
Operating system	Ubuntu 18.04 (64-bit)
Deep learning framework	Keras API using Tensorflow GPU back-end
Programming language	Python 3.6
CUDA compatibility	CUDA 10.0, CuDNN 7.5

C. Training of the Model

The proposed model is trained with Adam optimizer (having hyper parameters such as learning rate = 0.00001, first momentum decay = 0.9, and second momentum decay = 0.999.) for 50 number of epochs with batch size of 3 and temporal sliding window size of $T = 10$. The model loss of the trained model for both UCSD Ped1 and UCSD Ped2 are given in the Fig. 2 and Fig. 3 respectively.

D. Anomaly Detection using Regularity Score

The regularity score can be used to detect the anomalous frames as shown in the Fig. 4 and 5 corresponding to bicycle and cart as the anomalies respectively. The importance of threshold setting is very much crucial as low value of threshold will generate false alarms and high value will miss the real anomalies as shown in Fig. 6.

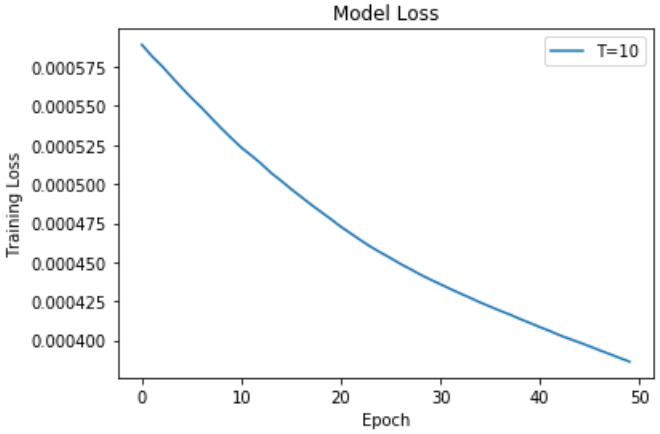

Fig. 2: Model loss for the UCSD Ped1.

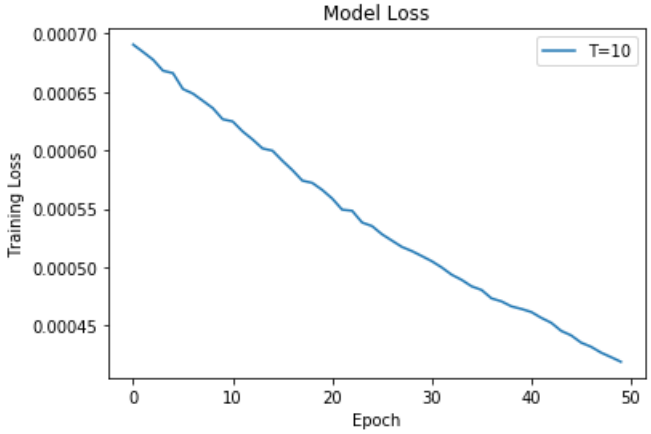

Fig. 3: Model loss for the UCSD Ped2.

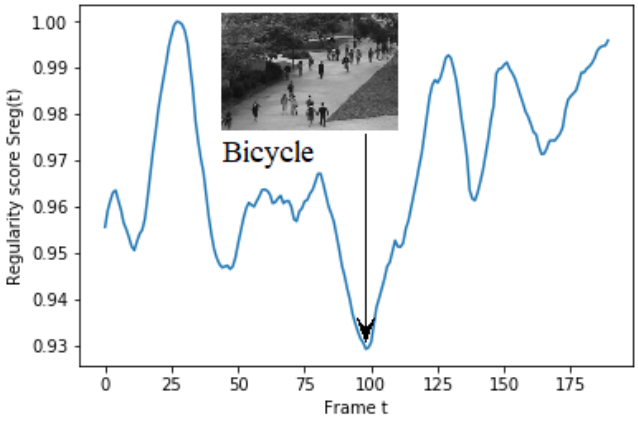

Fig. 4: Regularity score of video #1 of the UCSD Ped1 dataset.

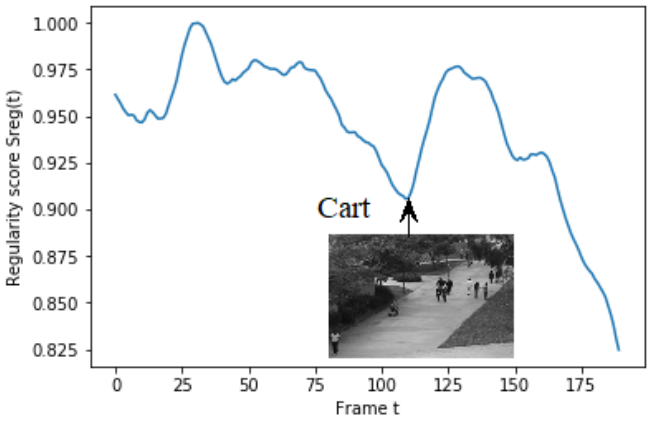

Fig. 5: Regularity score of video #23 of the UCSD Ped1 dataset.

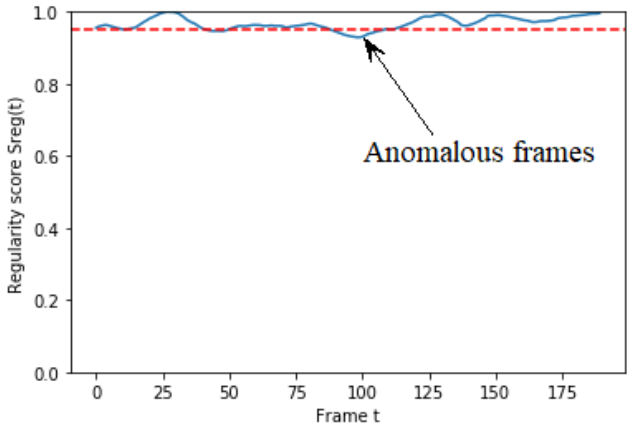

Fig. 6: Setting threshold value at 0.94 (less sensitive system)for video #1 of the UCSD Ped1 dataset.

E. Comparative Analysis

A comparative analysis of the proposed method with the state-of-art is presented in Table II and the results are found to be promising. Here, comparison is carried out in terms of two important performance parameters such as Area Under the Curve (AUC) and Equal Error Rate (EER). Both AUC and EER are evaluated based on the Receiver Operating Characteristics (ROC) curve (i.e., a two-dimensional plot between False Postive Rate on X-axis and True Positive Rate on Y-axis). The high values of AUC and low values of EER are preferred.

TABLE II: Comparative Analysis of the Proposed Method

Ref.	**Technique**	**UCSD Ped1**		**UCSD Ped2**	
		AUC (%)	**EER (%)**	**AUC (%)**	**EER (%)**
[5]	ConvAE	81.0	27.9	90.0	21.7
[6]	STAE	89.9	12.5	87.4	12.4
[18]	HOF Orientation	72.7	33.1	87.5	20.0
Proposed Method	ConvSTAE	**90.1**	**11.9**	88.3	**11.3**

V. Conclusions

The convolution spatiotemporal autoencoder based video anomaly detection technique with improved model architecture is successfully applied to the challenging bench-marked

datasets. Here, both spatial and temporal feature learning are used to detect the video anomaly based on the regularity score. The training is carried out in end-to-end layer fashion only with the normal data classes. Hence, the problem of anomalous dataset scarcity is addressed with this type of semi-supervised learning approach. Once the frames are detected as anomalous ones, they are stored as a separate video segment. In the later stage of processing, this separate video data segment can be classified using more complex deep supervised learning models for in-depth analysis.

REFERENCES

[1] B. Yang, J. Cao, R. Ni, and L. Zou, "Anomaly detection in moving crowds through spatiotemporal autoencoding and additional attention," *Advances in Multimedia*, vol. 2018, 2018.

[2] Y. Zhao, B. Deng, C. Shen, Y. Liu, H. Lu, and X.-S. Hua, "Spatio-temporal autoencoder for video anomaly detection," in *Proc. 25th ACM Int. Conf. on Multimedia*, 2017, pp. 1933–1941.

[3] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6479–6488.

[4] B. R. Kiran, D. M. Thomas, and R. Parakkal, "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos," *Journal of Imaging*, vol. 4, no. 2, p. 36, 2018.

[5] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2016, pp. 733–742.

[6] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *Int. Symp. on Neural Networks*. Springer, 2017, pp. 189–196.

[7] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proc. IEEE Int. Conf. on computer vision*, 2015, pp. 4489–4497.

[8] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2017, pp. 665–674.

[9] M. Ribeiro, A. E. Lazzaretti, and H. S. Lopes, "A study of deep convolutional auto-encoders for anomaly detection in videos," *Pattern Recognition Letters*, vol. 105, pp. 13–22, 2018.

[10] S. Sudhakaran and O. Lanz, "Learning to detect violent videos using convolutional long short-term memory," in *Proc. IEEE 14th Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, 2017, pp. 1–6.

[11] L. Wang, F. Zhou, Z. Li, W. Zuo, and H. Tan, "Abnormal event detection in videos using hybrid spatio-temporal autoencoder," in *Proc. IEEE 25th Int. Conf. on Image Processing (ICIP)*, 2018, pp. 2276–2280.

[12] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[13] H. Sellat, "Anomaly detection in videos using lstm convolutional autoencoder," https://towardsdatascience.com, Oct. 2019.

[14] V. Patraucean, A. Handa, and R. Cipolla, "Spatio-temporal video autoencoder with differentiable memory," *arXiv preprint arXiv:1511.06309*, 2015.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[16] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. on artificial intelligence and statistics*, 2010, pp. 249–256.

[17] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2010, pp. 1975–1981.

[18] T. Wang and H. Snoussi, "Histograms of optical flow orientation for abnormal events detection," in *Proc. IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, 2013, pp. 45–52.