Signal, Image and Video Processing
https://doi.org/10.1007/s11760-020-01740-1

ORIGINAL PAPER

Residual spatiotemporal autoencoder for unsupervised video anomaly detection

K. Deepak[1] · S. Chandrakala[1] · C. Krishna Mohan[2]

Received: 20 March 2020 / Revised: 21 June 2020 / Accepted: 6 July 2020

Abstract
Modeling abnormal spatiotemporal events is challenging since data belonging to abnormal activities are less in the course of a surveillance stream. We solve this issue using a normality modeling approach, where abnormalities are detected as deviations from the normal patterns. To this end, we propose a residual spatiotemporal autoencoder, which is trainable end-to-end to carry out the anomaly detection task in surveillance videos. Irregularities are detected using the reconstruction loss, where normal frames are reconstructed well with a low reconstruction cost, and the converse is identified as abnormal frames. We evaluate the effect of residual connections in the STAE architecture and presented good practices to train an autoencoder for video anomaly detection using benchmark datasets, namely CUHK-Avenue, UCSD-Ped2, and Live Videos. Comparisons with the existing approaches prove that the effectiveness of residual blocks is incremental than going deeper with additional layers to train a spatiotemporal autoencoder with good generalization across datasets.

Keywords Unsupervised anomaly detection · Surveillance videos · Residual connections · Spatiotemporal autoencoder · Real-time

1 Introduction

Recently, anomaly detection over surveillance videos has gained a lot of research interest due to its direct applicability in various domains such as video surveillance [23,28]. In an unsupervised setting, anomaly detection is carried out as an outlier detection problem. The aim is to learn a model of normal events using segments of normal activities from training data and detect events that deviate from the learned model while testing. The problem turns out to be more complicated when the data points exist in a high dimension. The existing approaches for abnormality detection use either handcrafted or deep features to characterize the spatiotemporal patterns. Ghrab et al. [6] used trajectory-based feature descriptors and performed hierarchical clustering to remove noise from the training data. Kaltsa et al. [13] extracted spatiotemporal cubes and obtained a merged feature vector comprising of Histograms of Oriented Gradients (HOG) for the spatial information and Histogram of Oriented Swarm Acceleration (HOSA) to capture temporal dynamics.

Deep networks can be used for the extraction of data-driven high-level descriptors. In deep feature learning methods [26,33], the image patches are represented using features learned from deep-neural networks and a one-class classifier such as one-class SVM is used to detect the anomalies. In an approach proposed by Hu et al. [10], a deep incremental slow feature analysis (D-IncSFA) was used as a pipeline network to extract features and detect anomalies.

The spatiotemporal deep autoencoders [8,20] are used to model high- dimensional data in an unsupervised/weakly supervised setting. The latent representation formed out of the encoder part acts as a compressed form of the input video segment, but contained with typical patterns of the data. In unsupervised anomaly detection, the autoencoder is trained on normal segments by minimizing their cost of reconstruction, and then, reconstruction cost is used as a threshold to detect anomalies. It is generally assumed that the reconstruc-

✉ S. Chandrakala
chandrakala@cse.sastra.edu ; sckala@cse.iitm.ac.in

K. Deepak
deepak@sastra.ac.in ; deepu6892@gmail.com

C. Krishna Mohan
ckm@iith.ac.in

[1] Intelligent Systems Group, School of Computing, SASTRA University, Thanjavur 613401, India

[2] Visual Learning and Intelligence Group, Department of Computer Science and Engineering, Indian Institute of Technology, Hyderabad, India

Published online: 20 July 2020

tion error will be lower for the normal segments since they are close to the training data, while the reconstruction error becomes higher for the abnormal segments [7,8]. Recently, Hasan et al. [8] propose to learn temporal regularities in videos using a 2D convolutional autoencoder. It can learn both the feature representation and the temporal visual characteristics of regular patterns in an end-to-end manner, and it is also claimed to be computationally more efficient than approaches that use sparse coding on large video datasets.

To obtain more accurate detection of abnormal visual patterns, we develop a residual spatiotemporal autoencoder (R-STAE), which is end-to-end trainable. Given the video segments of normal activities as training data, the proposed R-STAE performs unsupervised learning of the spatiotemporal representation of normal patterns and reconstruct them with low errors. In summary, the proposed R-STAE uses residual blocks to mitigate the vanishing gradient problem. Implemented in an end-to-end manner, the R-STAE provides promising performance for spatiotemporal abnormality detection.

2 Related work

Due to the contextually varying characteristics of abnormal events, most of the approaches deal with anomaly by considering them as outliers by modeling the normality that persists in the training data. A two-stage algorithm based on K-means clustering and OCSVM [11] was proposed to eliminate outliers by clustering the spatiotemporal cubes of normal activities. A generative model that captures temporal patterns was built using a fully connected autoencoder [8] which learns from local spatiotemporal features. Liu et al. [18] used a generative U-Net [25] and an optical Flow-Net [5] to predict the consecutive frame and thereby determine the anomaly by comparing it with the original frame. Revathi et al. [24] proposed to use an object tracking approach for feature extraction and a deep learning classifier to detect anomalies in videos.

Detecting anomalous events largely depends on the temporal ordering of its video structure. Del Giorno et al. [4] break this dependency by computing spatiotemporal descriptors which are shuffled among several classifiers. The scores of these classifiers are aggregated to determine the anomaly score. Recently, long short-term memory (LSTM) is used for addressing various tasks in the domains of speech recognition, natural language processing, and action recognition. Swathikaran Sudhakaran et al. used the convolutional long short-term memory [27], a variant of LSTM for aggregating frame-level features from a video to detect violent activities.

Tudor Ionescu et al. [30] proposed a framework that used a binary classifier that is trained iteratively to differentiate between two continuous video segments and remove the most discriminant features. A real-time end-to-end trainable two-stream network [1] was proposed for action detection. In this approach, optical flow is computed using FlowNet [5] which is fed as input to the motion stream. Early fusion is applied by concatenating the activations from both streams, and the whole network is trained end to end. Gong et al. [7] introduced a memory-augmented autoencoder which updates memory elements representing the normal characteristics of the input data. The intuition behind the Gong et al. [7] method is based on the fact that the model can sometimes learn to be more generalized and hence reconstruct abnormal segments significantly well. This will make the discrimination between normal and abnormal segments difficult in the testing phase.

3 Residual spatiotemporal autoencoder for detecting unusual events in surveillance videos

Powerful deep learning architectures that effectively capture the variations between anomalous and normal activities are considered to be of prime importance in case of detecting anomalies in unconstrained videos. Being a data-driven approach, deep models help in learning more generalized patterns that cover intra-class variations prevalent in various normal activities. In a recent method [3], they used two-stream residual networks for action recognition. As opposed to this, we propose to use a single stream residual spatiotemporal autoencoder (R-STAE) architecture to detect unusual events in surveillance videos as shown in Fig. 1.

3.1 Normality modeling using residual spatiotemporal autoencoder (R-STAE)

The aim is to extract spatiotemporal representations that can distinguish normal and abnormal events in video segments given as inputs. We propose to use residual spatiotemporal autoencoder (R-STAE) which consists of 3D convolution, deconvolution, and Conv.LSTM layers to learn patterns of normal activities from surveillance videos. Recent advancements in deep learning approaches enable the autoencoders to effectively encode any given data distribution with minimal loss of information.

The residual spatiotemporal autoencoder shown in Fig. 1 consists of eight layers with four layers in the encoder and decoder part each. The encoder part comprises of three 3D convolution layers with 256, 128, and 64 units, respectively. The convolution layers are used to extract spatial information from the given input video segments. The convolution operation is a matrix multiplication between the filter and the image patches with the help of a sliding window. Since ReLU's activation values have no upper bounds, hyperbolic tangent (tanh) is used as activation to ensure the property of symme-

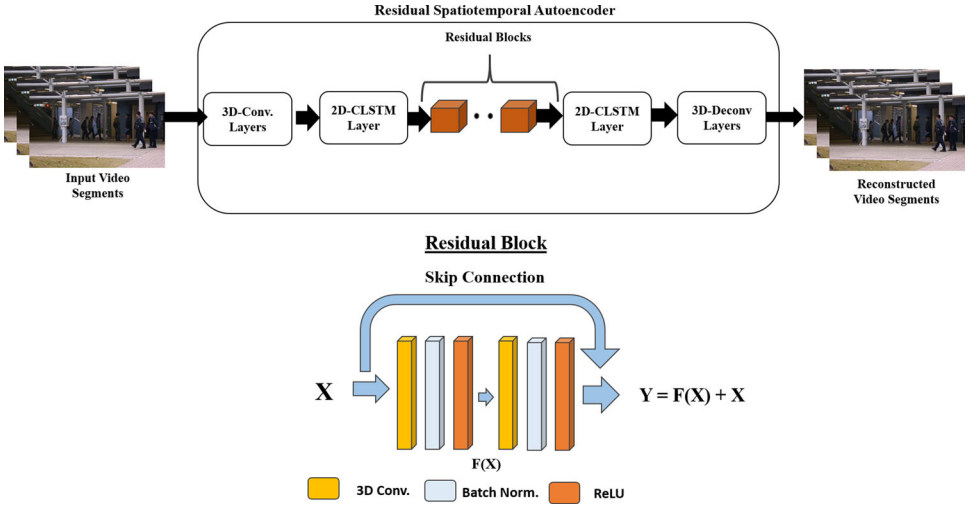

Fig. 1 Training residual spatiotemporal autoencoder for video anomaly detection

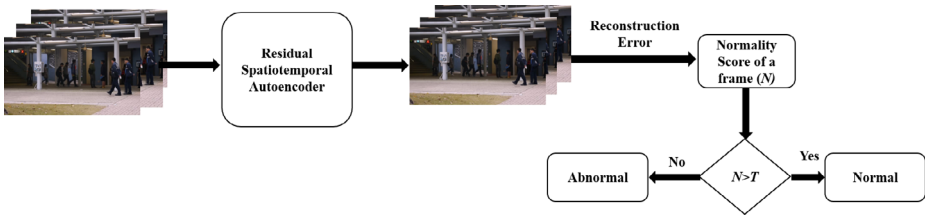

Fig. 2 Testing phase of residual spatiotemporal autoencoder for video anomaly detection

try in encoder and decoder functions. Batch normalization (BN) is employed as one of the regularization techniques to improve the training efficiency of the R-STAE. Conv.LSTM [32] layer is used in both the encoder and decoder parts of the R-STAE, respectively. Simple LSTMs cannot retain appearance information of video sequences. Thus, to capture the temporal dynamics of video sequences along with spatial information, Conv.LSTM was introduced where all the states are 3D tensors and can accommodate spatial dimensions. Let x be the input sequence at time step t, h is the hidden state, and gates are given by i, f, o with the cell output C. The convolution operation is given by $\star$, $\circledcirc$ is the Hadamard product, W denotes the weight matrices, and bias vectors are given by b. As provided in [32], Conv.LSTM is given by:

$$i_t = \sigma(W_i \star [x_t, h_{t-1}] + W_i \circledcirc C_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_f \star [x_t, h_{t-1}] + W_f \circledcirc C_{t-1} + b_f) \quad (2)$$

$$o_t = \sigma(W_o \star [x_t, h_{t-1}] + W_o \circledcirc C_t + b_o) \quad (3)$$

$$C_t = f_t \circledcirc C_{t-1} + i_t \circledcirc \tanh(W_c \star [x_t, h_{t-1}] + b_c) \quad (4)$$

$$h_t = o_t \circledcirc \tanh(C_t) \quad (5)$$

Apart from the Conv.LSTM layer, the decoder part consists of deconvolutional layers [34] also known as the convolutional transpose layers used for reconstruction. The hyper-parameters such as kernel size, the number of kernels, and strides were determined empirically beforehand while the kernel values are allowed to be initialized randomly.

Residual networks Motivated by the recent action recognition approaches [3,12], we utilize Residual Networks [9] (ResNet) to overcome the vanishing gradients problem prevalent in deep networks. The residual blocks used in our architecture are shown in Fig. 1. A basic residual block in a residual network contains an identity skip connection besides the existing convolution layers. This helps in propagating the information from the previous layers and also contributes to gradient flow during backpropagation, thus controlling the vanishing gradient problem. The equation of a residual block with input x is given by,

$$y = F(x) + x \quad (6)$$

The training is performed based on the reconstruction loss:

$$e = \| V_i - \hat{V}_i \|_2 \quad (7)$$

where V_i is the input video segment and $\hat{V}_i$ is the reconstructed video segment. Video segments are given to the R-STAE from which a model of normal activities is learned while reconstructing the given input. The testing is carried out as shown in Fig. 2. The average squared difference between the reconstructed frame and the actual frame is computed using the mean squared error (MSE). This is because the MSE values will be less for the normal frame and higher for an abnormal frame (since the model is trained for normality). Next, the normality score for a frame is obtained with

Table 1 Architecture of the proposed R-STAE

Layer	Output-Map Dim.	Kernel	Stride	Output channel
Image	112 × 112 × 8	–	–	–
Conv-3D 1 (tanh)	56 × 56 × 4	4 × 4 × 4	2	256
Conv-3D 2 (tanh)	28 × 28 × 2	4 × 4 × 4	2	128
Conv-3D 3 (tanh)	14 × 14 × 1	4 × 4 × 4	2	64
Conv.LSTM (Conv)	14 × 14 × 1	2 × 2	1	64
Residual Block 1 (2 layers with skip connections)				
Conv-3D 4 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	1	**64**
Conv-3D 5 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	**1**	**64**
Residual Block 2 (2 layers with skip connections)				
Conv-3D 6 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	1	**64**
Conv-3D 7 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	**1**	**64**
Residual Block 3 (2 layers with skip connections)				
Conv-3D 8 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	1	**64**
Conv-3D 9 (tanh)	**14 × 14 × 1**	**3 × 3 × 1**	**1**	**64**
Conv.LSTM (De-Conv)	14×14×1	2×2	1	64
DeConv-3D 1(tanh)	28×28×2	4×4×4	2	64
DeConv-3D 2(tanh)	56×56×4	4×4×4	2	128
DeConv-3D 3(tanh)	112×112×8	4×4×4	2	256

Bold indicates the highest result achieved for the corresponding approach/configuration

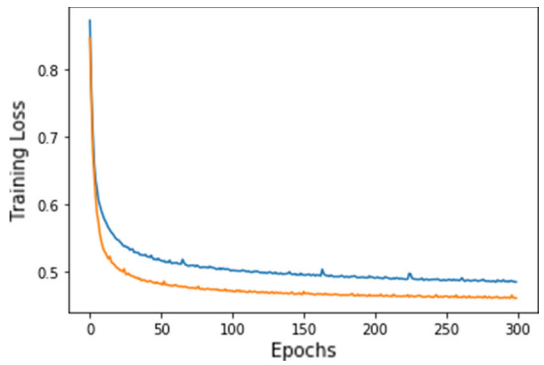

Fig. 3 Reconstruction loss for LV (Live Videos) dataset without residual blocks (Blue), reconstruction loss with residual blocks (Orange) (color figure online)

the formula,

$$\text{normscore} = 1 - (\text{MSE} - \min(\text{MSE}))/\max(\text{MSE})) \quad (8)$$

Normality scores for the total number of testing frames are computed, and it will be in the range [0–1]. A threshold value is empirically chosen as 0.7, which implies that if the normality score is greater than 0.7, the test example is labeled as normal and the converse is abnormal.

The aim is to achieve a meaningful reconstruction of the normal video segments. To achieve this, the reconstruction loss during the training phase has to be decreased through architectural stability. No pre-trained network has been used in the training phase for the purpose of feature extraction. The input to the R-STAE network is a video segment comprised of a sequence of stacked grey-scale images rather than images with RGB channels. This helps in alleviating the burden of reconstructing redundant information in the frames during testing. The dimension of the input video segment is 112 * 112 * 1 * 8, where 1 indicates the number of channels in the image, and 8 is the number of continuous frames forming a video segment. Data augmentation does not help much in reducing reconstruction loss during training.

The overall architecture of the proposed R-STAE is shown in Table 1. The input to the R-STAE network is a video segment comprising of eight continuous frames with a resolution of 112 * 112. The transformed size of feature maps in every layer including residual layers is also presented in Table 1. Our R-STAE architecture consists of three convolution layers, one Conv.LSTM layer in the encoder part and three deconvolution layers, one Conv.LSTM (DeConv) layer in the decoder part with residual blocks placed in-between encoder and decoder blocks. Max Pooling layers are not used to avoid the loss of spatial information in the input frames. The effect of adding residual blocks to the R-STAE network is observed in Fig. 3. When studied on LV dataset, it can be observed that adding residual blocks to the network helps in achieving low reconstruction loss when compared to that of the network with no residual blocks.

4 Experimental studies

4.1 Datsasets used

The datasets used for the experiments are designed for unsupervised modeling (i.e) the training data contain only normal videos while the testing data will contain both normal and abnormal videos. In CUHK Avenue [19] dataset, there are a total of 16 training and 21 testing clips each of duration not more than two minutes. A total of 15,328 and 15,324 frames are present in training and testing, respectively. The resolution of each frame is 360 * 640, and the frame rate for the video clips is 25 frames per second (fps). The Live Videos (LV) [17] dataset consists of 30 videos in total each of which is a unique scenario containing both training/testing sequences. The frame rate varies from 7.5 to 30 frames per second with a minimum resolution of 176 * 144 and a maximum of 1280 * 720. UCSD Ped2 [21] dataset is composed of 16 training and 12 testing videos with a resolution of 240 ∗ 360 pixels. Some abnormal events in the above- mentioned datasets include throwing objects, pedestrian walk way anomalies, violence and robbery, etc.

4.2 Implementation

4.2.1 Preprocessing and training

Each frame in an input video is resized to a resolution of 112 * 112. A set of eight consecutive frames are grouped to form video segments of constant duration. The R-STAE architecture is implemented on Keras deep learning framework. Experiments are conducted on NVIDIA Quadro P5000 GPU. The autoencoder is optimized with Adam optimizer [15], which is a simple and efficient approach. tanh activation was used in the 3D Convolutional layers as tanh provides nonlinearity and effectively learns the underlying patterns. The dropout values are empirically chosen in the Conv.LSTM layers to avoid the problem of over-fitting. The model was trained batch-wise with a batch size of 16. The reconstruction time of a frame (on one Quadro P5000 GPU) is **0.07s**, detection time is **0.05s**, and total time is **0.12s**. To evaluate the performance of the proposed approach over the above-mentioned datasets, its corresponding AUC scores are used.

4.3 Performance analysis

Comparisons are made among existing state-of-art methods and our proposed approach as shown in Tables 2, 3, and 4 for the datasets Avenue [19], LV [17], and UCSD-Ped2 [21], respectively. For Avenue dataset, Hasan et al. [8] have utilized an end-to-end framework which uses convolutional autoencoder with standard HOG, HOF, and raw videos as inputs to learn the temporal regularity in video sequences and the model is trained using the reconstruction loss. A convolutional winner-take-all autoencoder (CONV-WTA) that takes only optical flow sequences to model normal events was proposed by Hanh et al. [29]. Instead of reconstruction loss, this approach utilizes OC-SVM to detect anomalies. As shown in Table 2, the proposed deep R-STAE approach is comparable to [7,31] and outperforms other state-of-the-art methods. The sharp increase in the detection performance of the MemAE [7] method is due to the reason that they have used a separate memory module to store the prototypical normal patterns which are then used to reconstruct the input video segments.

Even though the LV dataset is highly challenging due to its varying contextual nature, we have achieved significant improvement when compared to other methods applied over this dataset as shown in Table 3. The rejection of motion outlier [14] approach incorporated a hardware friendly approach with the help of KUGDA (Univariate Gaussian Discriminant Analysis) for anomaly detection. Even though the least frame processing time is achieved by the Biswas et al. [2] method, they did not use any methods such as optical flow or background subtraction methods to derive the motion fea-

Table 2 Performance over Avenue dataset

S. no	Method	AUC
1	Conv-Autoencoder [8]	0.70
2	Discriminative framework [4]	0.78
3	STAE-Grayscale [35]	0.77
4	STAE-optflow [35]	0.81
5	Sparse combination learning [19]	0.81
6	Conv-WTA+SVM [29]	0.82
7	MemAE [7]	0.83
8	Wang et al. [31]	0.85
9	**STAE**	**0.79**
10	**R-STAE**	**0.82**

Bold indicates the highest result achieved for the corresponding approach/configuration

Table 3 Performance over LV dataset

S. no	Method	AUC
1	Sparse dictionary [19]	0.11
2	H.264 [2]	0.15
3	Binary features [16]	0.18
4	K-Means with BS [14]	0.25
5	KUGDA with BS [14]	0.26
6	Conv-Autoencoder [8]	0.34
7	Conv.LSTM-Autoencoder [20]	0.39
8	**STAE**	**0.61**
9	**R-STAE**	**0.63**

Bold indicates the highest result achieved for the corresponding approach/configuration

Fig. 4 Reconstruction of frames belonging to UCSD-PED2 dataset. Top row: original frames. Bottom row: reconstructed frames using R-STAE

Table 4 Performance over UCSD-Ped 2 dataset

S. no	Method	AUC
1	Social force [22]	0.56
2	MPPCA+Social force [21]	0.69
3	Unmasking [30]	0.82
4	**Conv.Autoencoder** [8]	**0.90**
5	**MemAE** [7]	**0.94**
6	STAE	0.78
7	R-STAE	0.83

Bold indicates the highest result achieved for the corresponding approach/configuration

Table 5 Results for different numbers of residual blocks in the R-STAE

Residual blocks	Avenue (AUC)	LV (AUC)	Ped2 (AUC)
Without resnet	0.79	0.61	0.78
1	0.79	0.61	0.81
2	0.81	0.62	0.81
3	**0.82**	**0.63**	**0.83**
4	0.82	0.63	0.83

Bold indicates the highest result achieved for the corresponding approach/configuration

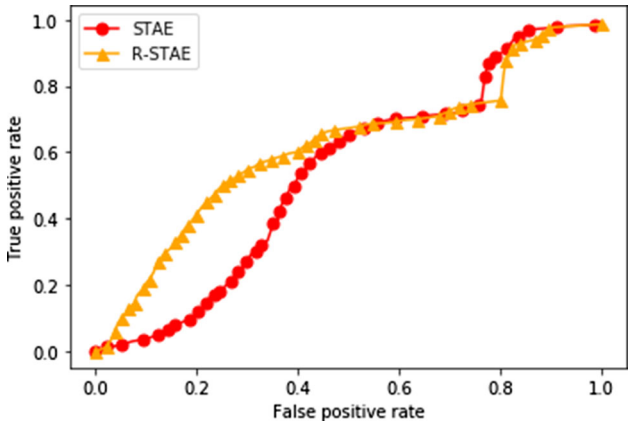

Fig. 5 Comparison of frame-wise ROC curve for STAE and R-STAE approaches on the LV dataset (Prediction scores are sampled alternatively with $N = 4$ to obtain better clarity of the curve)

tures which resulted in a major compromise of performance. Since each video contains a different scenario, it demands a generalized framework that can operate in a variety of environments. The R-STAE approach significantly outperforms the state-of-the-art techniques. With the help of confusion matrices, it can be observed that the number of false posi-

Table 6 Results for different sizes of hidden units in Conv.LSTM layer of R-STAE

Units in C.LSTM	Network parameters	Avenue (AUC)	LV (AUC)	Ped2 (AUC)
16	4,929,313	0.76	0.57	0.81
32	5,155,329	0.81	0.60	0.81
64	5,764,033	**0.82**	**0.63**	**0.83**
128	7,608,129	**0.82**	0.62	**0.83**

Bold indicates the highest result achieved for the corresponding approach/configuration

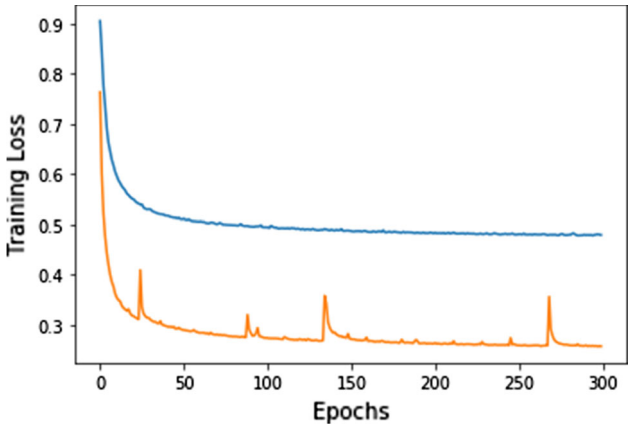

Fig. 6 Training loss on UCSD-Ped2 dataset—tanh (Orange) versus ReLU (Blue) activation functions (color figure online)

tives in the R-STAE approach is comparatively less than that of the STAE without residual blocks.

When compared to other datasets, UCSD-Ped2 is a relatively small dataset and less complex in nature. The proposed R-STAE approach improves the frame-level detection performance by about 14% when compared with the MPPCA+Social Force [21] approach. Still, the R-STAE achieves only comparable results when compared to other state-of-the-art approaches given in Table 4. One possible justification for slightly degraded performance of the R-STAE in UCSD-Ped2 would be due to a smaller number of training examples present in the UCSD Ped2 dataset. The sparse regularization technique and addition of memory module seem to be steadily beneficial for the anomaly detection task since the MemAE [7] approach consistently proves its significance in both Avenue and UCSD Ped2 datasets. Figure 4 shows a visual analysis of detection performance over the UCSD ped2 dataset. Unusual objects such as vehicles move comparably faster than normal objects such as pedestrians. Therefore, abnormal moving objects in the frames are blurry because of the high reconstruction error. Overall, the normality scores corresponding to segments of normal events are higher than those of unusual events which proves the significance of the proposed R-STAE approach.

4.4 Ablation studies

In this section, we compare the effects of the following: (1) the influence of residual blocks in detection performance, (2) Conv.LSTM layer with varying dimensions, and (3) the effect of tanh activation function against the ReLU activation function. The influence of residual blocks added to the network is presented in Table 5 and Fig. 5. Three residual blocks improve the accuracy of up to 3% for all datasets, whereas adding the residual blocks higher than three showed no improvement. From Fig. 5, it is also observed that the R-STAE approach significantly achieved a better area under the curve when compared to STAE architecture without residual blocks.

To ensure the reconstruction ability of the R-STAE architecture, the hidden units of the Conv.LSTM layer is chosen empirically to achieve a trade-off between performance and the total network parameters. The effect of the number of hidden units assigned to the Conv.LSTM layer is presented in Table 6. The number of hidden units in the Conv.LSTM layer is used to form a compressed representation of a video segment. A minimum number of hidden units may lead to more loss of information, whereas a large number of hidden units in the Conv.LSTM layer might introduce redundancy in the latent representation. In our experiments, the convolution LSTM layer with 64 hidden units provides better reconstruction for all three datasets. Another observation on the ablation study is presented in Fig. 6. It is seen that the usage of the tanh activation function helps in achieving lower training loss when compared to ReLU for the R-STAE network over the UCSDPed2 dataset.

5 Conclusion

Recently, spatiotemporal autoencoder-based approaches are promising in detecting anomalous activities in surveillance videos. We propose to use an end-to-end residual spatiotemporal autoencoder (R-STAE) for unusual event detection in videos. Our experiments on various benchmark datasets show that the proposed architecture is able to perform frame-level abnormality detection quite well with the help of residual blocks. Three residual blocks along with Conv.LSTM layers in the proposed R-STAE architecture provides consistently better detection performance. The results against some state-of-the-art methods proved the effectiveness of proposed R-STAE architecture.

Acknowledgements The authors would like to acknowledge the following funding agencies: "Council of Scientific and Industrial Research (CSIR)" (09/1095(0043)/19-EMR-I) and (No.DST/CSRI/2017/131(G)) project under the "Cognitive Science Research Initiative (CSRI)" sanctioned by the Department of Science and Technology, Government of India.

References

1. Ali, A., Taylor, G.W.: Real-time end-to-end action detection with two-stream networks. In: 2018 15th Conference on Computer and Robot Vision (CRV), IEEE, pp. 31–38 (2018)
2. Biswas, S., Babu, R.V., (2013) Real time anomaly detection in h. 264 compressed videos. In: Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), pp. 1–4. IEEE (2013)

3. Christoph, R., Pinz, F.A.: Spatiotemporal residual networks for video action recognition. Advances in Neural Information Processing Systems, pp. 3468–3476 (2016)
4. Del Giorno, A., Bagnell, J.A., Hebert, M.: A discriminative framework for anomaly detection in large videos. In: European Conference on Computer Vision, pp. 334–349. Springer (2016)
5. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2758–2766 (2015)
6. Ghrab, N.B., Fendri, E., Hammami, M.: Abnormal events detection based on trajectory clustering. In: 2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV), pp. 301–306. IEEE (2016)
7. Gong, D., Liu, L., Le, V., Saha, B., Mansour, MR., Venkatesh, S., Hengel, Avd.: Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection (2019). arXiv preprint arXiv:1904.02639
8. Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, AK., Davis, L.S.: Learning temporal regularity in video sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 733–742 (2016)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. Hu, X., Hu, S., Huang, Y., Zhang, H., Wu, H.: Video anomaly detection using deep incremental slow feature analysis network. IET Comput. Vis. **10**(4), 258–267 (2016)
11. Ionescu, R.T., Smeureanu, S., Popescu, M., Alexem B.: Detecting abnormal events in video using narrowed normality clusters. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1951–1960. IEEE (2019)
12. Iqbal, A., Richard, A., Kuehne, H., Gall, J.: Recurrent residual learning for action recognition. In: German Conference on Pattern Recognition, pp. 126–137. Springer (2017)
13. Kaltsa, V., Briassouli, A., Kompatsiaris, I., Strintzis, M.G.: Swarm-based motion features for anomaly detection in crowds. In: 2014 IEEE International Conference on Image Processing (ICIP), pp. 2353–2357. IEEE (2014)
14. Khan, M.U.K., Park, H.S., Kyung, C.M.: Rejecting motion outliers for efficient crowd anomaly detection. IEEE Trans. Inf. Forensics Secur. **14**(2), 541–556 (2018)
15. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014). arXiv preprint arXiv:1412.6980
16. Leyva, R., Sanchez, V., Li, C.T.: Abnormal event detection in videos using binary features. In: 2017 40th International Conference on Telecommunications and Signal Processing (TSP), pp. 621–625. IEEE (2017)
17. Leyva, R., Sanchez, V., Li, C.T.: The LV dataset: a realistic surveillance video dataset for abnormal event detection. In: 2017 5th International Workshop on Biometrics and Forensics (IWBF), pp 1–6. IEEE (2017)
18. Liu, W., Luo, W., Lian, D., Gao, S.: Future frame prediction for anomaly detection—a new baseline. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6536–6545 (2018)
19. Lu, C., Shi, J., Jia, J.: Abnormal event detection at 150 fps in matlab. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2720–2727 (2013)
20. Luo, W., Liu, W., Gao, S.: Remembering history with convolutional LSTM for anomaly detection. In: 2017 IEEE International Conference on Multimedia and Expo (ICME), pp. 439–444. IEEE (2017)
21. Mahadevan, V., Li, W., Bhalodia, V., Vasconcelos, N.: Anomaly detection in crowded scenes. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition pp. 1975–1981. IEEE (2010)
22. Mehran, R., Oyama, A., Shah, M.: Abnormal crowd behavior detection using social force model. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 935–942. IEEE (2009)
23. Noceti, N., Odone, F., Sciutti, A., Sandini, G.: Exploring biological motion regularities of human actions: a new perspective on video analysis. ACM Trans. Appl. Percept. **14**(3), 21:1–21:20 (2017). https://doi.org/10.1145/3086591
24. Revathi, A., Kumar, D.: An efficient system for anomaly detection using deep learning classifier. Signal Image Video Process. **11**(2), 291–299 (2017)
25. Ronneberger, O. Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 234–241. Springer (2015)
26. Sabokrou, M., Fayyaz, M., Fathy, M., Klette, R.: Deep-cascade: cascading 3d deep neural networks for fast anomaly detection and localization in crowded scenes. IEEE Trans. Image Process. **26**(4), 1992–2004 (2017)
27. Sudhakaran, S., Lanz, O.: Learning to detect violent videos using convolutional long short-term memory. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2017)
28. Tadros, T., Cullen, N.C., Greene, M.R., Cooper, E.A.: Assessing neural network scene classification from degraded images. ACM Trans. Appl. Percept. **16**(4), 21:1–21:20 (2019). https://doi.org/10.1145/3342349
29. Tran, H.T., Hogg, D.: Anomaly detection using a convolutional winner-take-all autoencoder. In: Proceedings of the British Machine Vision Conference 2017. British Machine Vision Association (2017)
30. Tudor Ionescu, R., Smeureanu, S., Alexe, B., Popescu, M.: Unmasking the abnormal events in video. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2895–2903 (2017)
31. Wang, S., Zeng, Y., Liu, Q., Zhu, C., Zhu, E., Yin, J.: Detecting abnormality without knowing normality. In: ACM International Conference on Multimedia. ACM Press (2018)
32. Xingjian, S., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.c.: Convolutional lstm network: a machine learning approach for precipitation nowcasting. In: Advances in Neural Information Processing Systems, pp. 802–810 (2015)
33. Xu, D., Ricci, E., Yan, Y., Song, J., Sebe, N.: Learning deep representations of appearance and motion for anomalous event detection (2015). arXiv preprint arXiv:1510.01553
34. Zeiler, M.D., Krishnan, D., Taylor, G.W., Fergus, R.: Deconvolutional networks. In: Cvpr, vol. 10, p. 7 (2010)
35. Zhao, Y., Deng, B., Shen, C., Liu, Y., Lu, H., Hua, X.S.: Spatio-temporal autoencoder for video anomaly detection. In: ACM Multimedia (2017)