Hindawi
Scientific Programming
Volume 2021, Article ID 6412608, 8 pages
https://doi.org/10.1155/2021/6412608

Research Article

Abnormal Event Detection in Videos Based on Deep Neural Networks

Qinmin Ma

School of Artificial Intelligence, Shenzhen Polytechnic, Shenzhen 518055, China

Correspondence should be addressed to Qinmin Ma; mqm@szpt.edu.cn

Received 8 July 2021; Revised 24 July 2021; Accepted 30 July 2021; Published 5 August 2021

Academic Editor: Bai Yuan Ding

Abnormal event detection has attracted widespread attention due to its importance in video surveillance scenarios. The lack of abnormally labeled samples makes this problem more difficult to solve. A partially supervised learning method only using normal samples to train the detection model for video abnormal event detection and location is proposed. Assuming that the distribution of all normal samples complies to the Gaussian distribution, the abnormal sample will appear with a lower probability in this Gaussian distribution. The method is developed based on the variational autoencoder (VAE), through end-to-end deep learning technology, which constrains the hidden layer representation of the normal sample to a Gaussian distribution. Given the test sample, its hidden layer representation is obtained through the variational autoencoder, which represents the probability of belonging to the Gaussian distribution. It is judged abnormal or not according to the detection threshold. Based on two publicly available datasets, i.e., UCSD dataset and Avenue dataset, the experimental are conducted. The results show that the proposed method achieves 92.3% and 82.1% frame-level AUC at a speed of 571 frames per second on average, which demonstrate the effectiveness and efficiency of our framework compared with other state-of-the-art approaches.

1. Introduction

With the development of chip technology and cost reeducation of bandwidth and storage equipment cost, etc., network digital cameras have replaced traditional analog cameras and are widely deployed in museums, banks, airport, etc. In order to strengthen public safety protection and prevent crime, the video surveillance has entered the era of blowout. According to HIS Data Display [1], the new video surveillance cameras installed in 2016 worldwide will produce approximately 566 GB of data in one day. To 2023, the data amount is estimated to reach 3500 GB. The rapid growth of video data puts forward higher requirements for video understanding. Intelligent surveillance technology has replaced traditional video surveillance personnel to achieve real-time structured processing and analysis of massive video data. As one of the key technologies of intelligent monitoring technology, abnormal event detection is from real-time detection in massive surveillance video data, which are a small number of abnormal events that are inconsistent with most normal events.

In recent years, abnormal event detection has gradually become a research hotspot in the field of computer vision and pattern recognition. The main difficulty is that the scenes of abnormal events are diverse. It is difficult to define an interface covering the boundaries of various possible abnormal events. A common solution is to define an abnormal event as a low probability event relative to a normal event, which enables statistical processing of abnormal events, deviated from expectations, and events that are inconsistent with normal samples are abnormal events. Same as the most popular ideas in the field of computer vision and pattern recognition, the existing methods for detecting abnormal events can be roughly divided into two steps [2–4]: event representation and anomaly detection model. Event representation is to extract appropriate features from the video to represent the event. Due to the ambiguity of event definition, the event can be characterized by object-level features or pixel-level features. The former often uses object trajectory features [5] or object appearance characteristics [6] (such as sports history images and sports energy images) to indicate an event. However, object-level features rely on detecting and tracking objects, which is difficult to handle in a crowded scene, especially for moving objects that block each other. For pixel-level features, they are often extracted from two-dimensional image blocks or three-dimensional video cubes to represent, such as spatiotemporal gradient (STG) [7], optical histograms of optical flow (HOF) [8, 9], and mixture of dynamic textures (MDT) [4]. After obtaining the characteristics that represent the event, the next question is to build an anomaly detection model. The anomaly detection model is to establish rules or models for normal events. Then, the test event that violates the rules or does not conform to the model is treated as an exception. Common models are cluster-based detection models [10], detection model based on state inference [11], and detection model based on sparse reconstruction [8, 12]. Among them, the cluster-based detection model clusters similar normal events together. Therefore, samples far away from these cluster centers during the testing phase are regarded as abnormal events. The state inference model assumes that normal events will undergo a fixed change over time. And, the abnormal event does not conform to this change. For detection models based on sparse reconstruction, the main principle is that the reconstruction of normal events has a small error relative to the reconstruction of abnormal events.

Although the above methods have achieved certain results in previous studies, there is a problem because the event representation and anomaly detection models are designed separately. Such operations cause researchers to spend too much effort to design them separately, but these methods often fail; when the video scene changes, generalization ability is poor. Recently, deep learning has achieved excellent results in the fields of computer vision and pattern recognition and intelligent manufacturing, such as object recognition [6, 13], object detection [14], behavior recognition [15], and health diagnosis. The key to the success of deep learning methods is that the two steps of feature representation and pattern recognition are jointly optimized, which can maximize the performance of the joint collaboration between them. It can further improve the generalization ability of the method for different scenarios. Driven by the success of deep learning technology, researchers began to apply it to abnormal event detection [16–18]. In [16], a three-channel architecture was proposed which used autoencoder on each channel (Autoencoder) [17]. To learn features, a single-class support vector machine (SVM) is employed afterwards to predict the anomaly score of each channel. Finally, the abnormal scores of the three channels are merged as the final basis for judging abnormalities. Sabokrou et al. introduced a cascaded anomaly detection method, which detected abnormal events based on the reconstruction error of the autoencoder and the sparsity of the sparse autoencoder. Based on manual features and short video clips, Hasan et al. adopted the fully connected autoencoder and fully convolution autoencoder to learn the time regularity of normal events. Then, according to the reconstruction error, the time regularity score of normal events was calculated to detect abnormalities. However, these methods are based on deep reconstruction treat samples that are different from normal samples as anomalies. It ignores the small probability of abnormal events. A large number of normal samples that did not appear are often misjudged as abnormal, leading to false alarms. Unlike these methods above, in this paper, we propose an end-to-end deep learning framework for abnormal event detection. The proposed method is based on variational autoencoder (VAE) [19–22], which can map high-dimensional raw input data to low-dimensional hidden layer representations through deep learning technology. And, it constrains the low-dimensional hidden layer representation to conform to a Gaussian distribution. Therefore, the hidden layer of the normal sample indicates that the probability value calculated for the Gaussian distribution is relatively large. The hidden layer of abnormal samples indicates that the probability value calculated for the Gaussian distribution will be relatively small. Actually, obtaining the hidden layer representation and constraining to a Gaussian distribution can, respectively, correspond to the two main steps of anomaly detection: event representation and anomaly detection model. In the proposed method, the two main steps are jointly optimized through an end-to-end deep learning framework, which can improve the generalization ability. Experimental results on two public datasets show that the proposed method has strong generalization ability and the detection performance reaches the level of current technology development.

2. VAE for Anomaly Detection

The overall process of the proposed method can be described as follows. During the training phase, the space-time cube of normal samples is densely sampled. The original pixels are directly used as the input of the VAE to learn the Gaussian distribution in the hidden layer representation of the input data. Then, for a test sample, the hidden layer representation of the test sample is obtained through the VAE, which calculates the probability that it belongs to the Gaussian distribution and uses it as an anomaly score. At last, the samples with abnormal scores below the threshold are judged to be abnormal. In this section, we first briefly introduce the principle of the autoencoder. Then, the proposed method of video abnormal event detection based on variational autoencoder is elaborated.

2.1. Principle of Autoencoder. Autoencoder [17] maps the input data to the hidden layer space to get its hidden layer representation. Through its hidden layer representation, the original input data can be reconstructed. Self-encoder by encoder $f_{\mathbf{w}_1}(\bullet)$ and decoder $g_{\mathbf{w}_2}(\bullet)$ composition can be expressed as

$$\begin{aligned}\mathbf{z} &= f_{\mathbf{w}_1}(\mathbf{x}),\\ \mathbf{x}' &= g_{\mathbf{w}_2}(\mathbf{z}),\end{aligned} \tag{1}$$

where $\mathbf{x}$ and $\mathbf{x}'$ represent the input of the autoencoder and the input of the reconstruction, respectively, $\mathbf{z}$ is the hidden representation for $\mathbf{x}$, and $\mathbf{W}_1$ and $\mathbf{W}_2$ are the parameters of the neural network. In order to minimize the input $\mathbf{x}$ and reconstruct the input $\mathbf{x}'$, the reconstruction error between is obtained as follows:

$$\min_{\mathbf{w}_1,\mathbf{w}_2} \|\mathbf{x}-\mathbf{x}'\|_2^2. \tag{2}$$

The hidden layer representation of the autoencoder is often used as effective features, which directly enter into the subsequent pattern recognition model. In order to improve the expressive ability of hidden layer representation, the noise reduction autoencoder [23] and sparse autoencoder [17] were developed by introducing noise and increasing sparsity constraints. The hidden layer representation is robust and sparsity against partial damage of data.

Suing error-based reconstruction [18, 24, 25] or directly extracting the hidden layer representation as a feature [16], autoencoders have been successfully used to solve anomaly detection tasks. However, these methods ignore the probability model in which normal samples occur with high probability and abnormal samples occur with low probability. To solve this problem, we assume that the hidden layer representation of the normal sample conforms to the Gaussian distribution, and a video abnormal event detection method based on variational autoencoder is proposed.

2.2. Anomaly Detection Model Based on VAE. Given n normal training samples $\mathbf{x} = \{x_i \in \mathbb{R}^s\}_{i=1}^n$, where the dimension of the sample is s, then the VAE [19–22] learns that the hidden layer represents the Gaussian distribution in the space. In the hidden layer representation space, assuming that the training samples conform to the Gaussian distribution, which means that all training samples are clustered into one cluster center, the samples far from the cluster center are abnormal samples.

Specifically, the hidden layer representation $\mathbf{z}$ satisfies

$$\mathbf{z} \sim \mathcal{N}(0,\mathbf{I}), \tag{3}$$

where $\mathbf{I}$ is the identity matrix. Similar to the reconstruction process of the autoencoder, VAE makes the data generated by the model very similar to the input data. Similar to the architecture of traditional autoencoders, VAE also includes two neural networks:

(1) Inferred network: a probabilistic encoder $q_\phi(\mathbf{z}|\mathbf{x})$ w will enter $\mathbf{x}$ mapped to hidden representation $\mathbf{z}$ close to reality posterior distribution $p(\mathbf{z}|\mathbf{x})$

(2) Generate network: a generative decoder $p_\theta(\mathbf{x}|\mathbf{z})$, which expresses the hidden layer without relying on any specific input prior $\mathbf{z}$ reconstruction to original training data $\mathbf{x}$

Among them, ϕ and θ represent the parameters of the two networks, respectively. Denote the network as "Encoder," the training data $\mathbf{x}$ is mapped to hidden layer representation $\mathbf{z}$. The generative network can be seen as "decoder," and the hidden layer $\mathbf{z}$ refactors to training data $\mathbf{x}$.

According to the theory of VAE [20], the loss function can be expressed as

$$\mathcal{L}(\theta,\phi,\mathbf{x}) = E_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\log\, p_\theta(\mathbf{x}|\mathbf{z})\right] - D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})\right. \tag{4}$$

In (5), the first item $\mathbf{x}$ is the expected log likelihood of the training data, which facilitates the decoder $p_\theta(\mathbf{x}|\mathbf{z})$ to rebuild training data $\mathbf{x}$. It can be considered as reconstruction error. When the reconstruction effect is good, the value of this item is smaller. According to the principle of Monte Carlo sampling, for each sample in the training data $\mathbf{x} = \{x_i \in \mathbb{R}^s\}_{i=1}^n$, for $q_\phi(\mathbf{z}|\mathbf{x})$ collection n a z_i, $1 \le i \le n$, there is

$$E_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\log\, p_\theta(\mathbf{x}|\mathbf{z})\right] \approx \frac{1}{n}\sum_{i=1}^{n}\log\, p_\theta(x_i|z_i), \tag{5}$$

where z_i is the hidden layer representation for x_i.

The second item is Kullback–Leibler divergence between $q_\phi(\mathbf{z}|\mathbf{x})$ and $p(\mathbf{z})$ [9], which represent the distribution that the encoder wants to learn and the prior distribution represented by the hidden layer. Kullback–Leibler divergence can measure the difference between two probability distributions. For two similar probability distributions, the Kullback–Leibler divergence is very small. Based on the hypothesis, $q_\phi(\mathbf{z}|\mathbf{x})$ is the normal distribution $\mathbf{N}(\mu,\delta)$, and there is

$$q_\phi(\mathbf{z}|\mathbf{x}) = \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}}\cdot e^{-(\mathbf{z}-\mathbf{u})^2/2\boldsymbol{\sigma}^2}. \tag{6}$$

According to (3), $p(\mathbf{z})$ can be further expressed as

$$p(\mathbf{z}) = \frac{1}{\sqrt{2\pi}}\cdot e^{-\mathbf{z}^2/2}. \tag{7}$$

According to (6) and (7), the second term of (4) can be expressed as

$$-D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}')\|p(\mathbf{z})\right) = -\int q_\phi(\mathbf{z}|\mathbf{x}')\log\frac{q_\phi(\mathbf{z}|\mathbf{x}')}{p(\mathbf{z})}\mathrm{d}\mathbf{z} = -0.5\left(1+\log\,\boldsymbol{\sigma}^2-\boldsymbol{\mu}^2-\boldsymbol{\sigma}^2\right). \tag{8}$$

Through the reparameterization method [24], the network parameters can be adjusted by (4), suing STD [26]. The VAE is essentially based on the autoencoder, which adds a Kullback–Leibler divergence. The hidden layer representation obtained by the encoder not only can reconstruct the input samples but also conforms to a Gaussian distribution.

Therefore, it is possible to detect abnormal events through the learned VAE.

2.3. Prediction. After learning the network weights of the VAE, for a test sample $\mathbf{y}$, the hidden layer representation $\mathbf{z}'$ from the inferred network $q_{\phi}(\mathbf{z}|\mathbf{x})$ can be obtained. According to (6), the probability of $\mathbf{z}'$ belonging to the Gaussian distribution is

$$p(\mathbf{z}') = \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}} \cdot e^{-(\mathbf{z}'-\mathbf{u})^2/2\boldsymbol{\sigma}^2}. \tag{9}$$

If the test sample $\mathbf{y}$ is a normal sample, it must appear in the high probability area of the Gaussian distribution. In contrast, the hidden layer of the abnormal sample indicates that the probability value calculated for the Gaussian distribution will be relatively small. Therefore, in order to infer whether the test sample is an abnormal sample, the threshold to make judgments can be set for $p(\mathbf{z}')$ as follows:

$$p(\mathbf{z}') \underset{\text{ab normal}}{\overset{\text{normal}}{\gtrless}} \delta, \tag{10}$$

where δ determines the threshold of the sensitivity of the detection method in this paper.

3. Experiment

In order to verify the effectiveness of the proposed method, experiments were conducted on two data sets, i.e., UCSD Ped1 dataset [8] and Avenue dataset [26]. And, the results are compared with several existing methods. Afterwards, we will introduce the experimental data, evaluation index, experimental details, and experimental results in detail.

3.1. Experimental Data and Evaluation Indicators. UCSD Ped1 dataset: the dataset records scenes on the sidewalk through a fixed camera, and the lens angle is slightly tilted. It contains 34 normal and 36 anomaly samples with the size of 238 × 158. Each video clip contains 200 frames. Normal events are pedestrians on the sidewalk. The abnormal events mainly include bicycles, skate, small car, and pedestrians walking on the lawn.

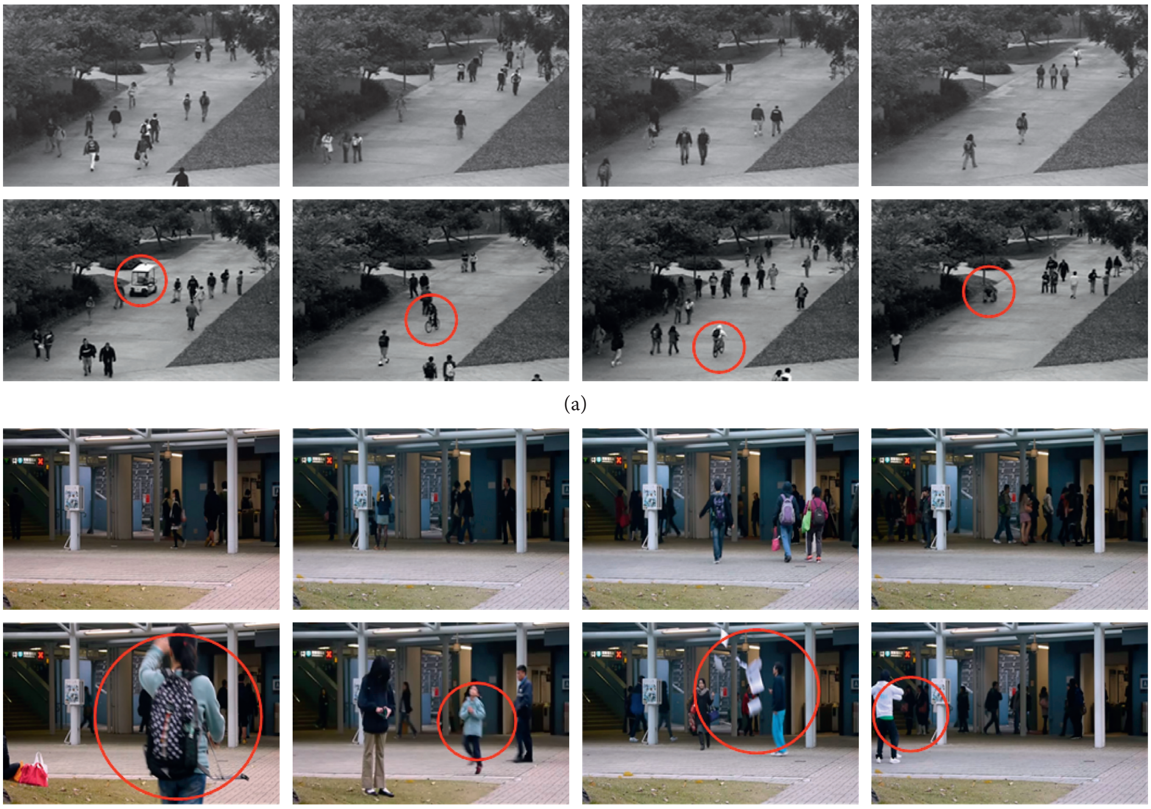
Avenue dataset: the dataset uses a fixed camera to record the scene in front of the school corridor and the lens angle is slightly tilted. It contains 15 normal and 21 anomaly samples with the size of 360 × 240. The dataset has a total of 30,652 frames. Normal events include pedestrians walking parallel to the camera. And, abnormal events include people running, throwing objects, and loitering. In Figure 1, some examples of events in two datasets are given, in which the upper pictures from each figure are normal ones while those on the bottom are anomalies.

Frame-level evaluation index and pixel-level evaluation index [11] are used to evaluate the performance of the detection method. For frame-level evaluation indicators, if a frame in the test sample contains at least one abnormal pixel, it is determined that the frame is an abnormal frame. For pixel-level evaluation indicators, if the anomalous area overlaps with the real anomaly marked area by more than 40%, it is determined that the frame is an abnormal frame. Whether it is a frame-level evaluation index or a pixel-level evaluation index, the detection rate (True Positive Rate, TPR) and false alarm rate (False Positive Rate, FPR) are calculated at first. Then, by changing the threshold δ in (10), the area under the curve (AUC) can be plotted.

3.2. Experimental Setup. For the two datasets, every frame is resized as 160 × 120. Each normal sample video clip is divided into the size of 10 × 10 × 5 with nonoverlapping space-time cubes. Then, these space-time cubes are converted into vectors with the size of 500 × 1 and normalized as the network input to train the weight of the variational autoencoder. In the proposed network, there are four hidden layers with 500, 500, 2000, and 30 neurons respectively. It uses a completely symmetrical network structure. The optimizer chooses the Adam Optimizer [8], and the initial learning rate is set to be 0.001. And, after every 1000 iterations, the learning rate reduces to 1/10 and the process stops at 10,000 iterations. The parameters are set as $\rho_1 = 0.9$ and $\rho_2 = 0.999$ and the batch size is 100. In the testing phase, the test video is also divided into sizes of 10 × 10 × 1 with nonoverlapping space-time cubes. They are input into the proposed network to obtain its hidden layer representation. Then, based on (10) whether the area is abnormal can be determined. The experimental hardware platform is NVIDIA GTX1070TI with video memory 8 GB. The software environment is Tensorflow and *Python*. In order to fully evaluate the performance of the proposed method, several comparison methods are drawn from current literatures, i.e., [7, 10, 17] and [22]. For simplicity, there are denoted as "Method 1," "Method 2," "Method 3," and "Method 4," respectively.

3.3. Results and Discussion. Figure 2 gives the results on the UCSD Ped1 dataset, where Figures 2(a) and 2(b) show the frame-level and pixel-level ROC curves. Figure 2 also provides the ROC curves of the proposed method and comparison ones. In the first three methods, the two steps of event representation and the establishment of the anomaly detection model are carried out separately. Among them, Method 1 extracts mixed dynamic texture features and then establishes a statistical inference anomaly detection model. Method 2 extracts spatiotemporal gradient features and then adopts sparse reconstruction method for anomaly detection. Method 3 uses autoencoder to extract features and single-class support vector machine for anomaly detection. Method 4 is an end-to-end deep learning method. The results of these four methods are obtained from the corresponding papers, among them Method 4 does not provide ROC curve.

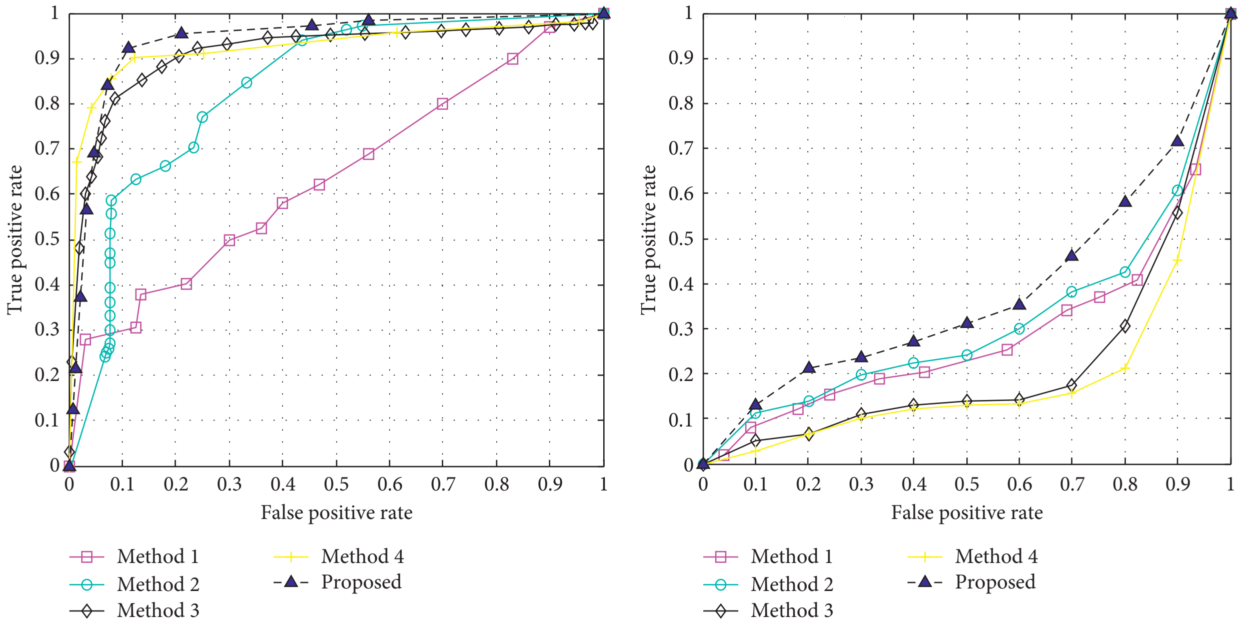
As can be seen from Figure 2, the proposed method achieved the best results on the frame-level evaluation criteria. On the pixel-level evaluation standard, the results of the proposed method are not much different from those of the other two methods, i.e., Method 2 and Method 3, but obviously better than that of Method 1. Table 1 shows the comparison results of different algorithms on the UCSD Ped1 dataset at the frame level and the pixel level. The proposed method achieves 92.3% frame-level AUC and



(a)

(b)

Figure 1: Examples of some events in the abnormal event detection dataset. (a)Examples from the UCSD Ped1 dataset. (b)Examples from the Avenue dataset.



(a)

(b)

Figure 2: ROC curves for the UCSD Ped1 dataset. (a)Frame-level ROC. (b) Pixel-level ROC.

71.4% pixel-level AUC, which are better than all other comparison methods. It is worth noting that learning temporal regularity is also an end-to-end deep learning method. However, the experimental results are clearly lower than the proposed method. This is because the method uses each frame of the video as the input of the neural network.

Table 1: Comparison with the existing methods in terms of AUC% for the UCSD Ped1 dataset.

Method type	Frame-level AUC (%)	Pixel-level AUC (%)
Proposed	93.1	66.4
Method 1	82.3	45.1
Method 2	92.2	64.1
Method 3	91.9	65.2
Method 4	82.3	63.7

Table 2: Comparison with the existing methods in terms of frame-level AUC% for the Avenue dataset.

Method type	Frame-level AUC (%)
Ours	82.5
Method 2	81.1
Method 4	78.6

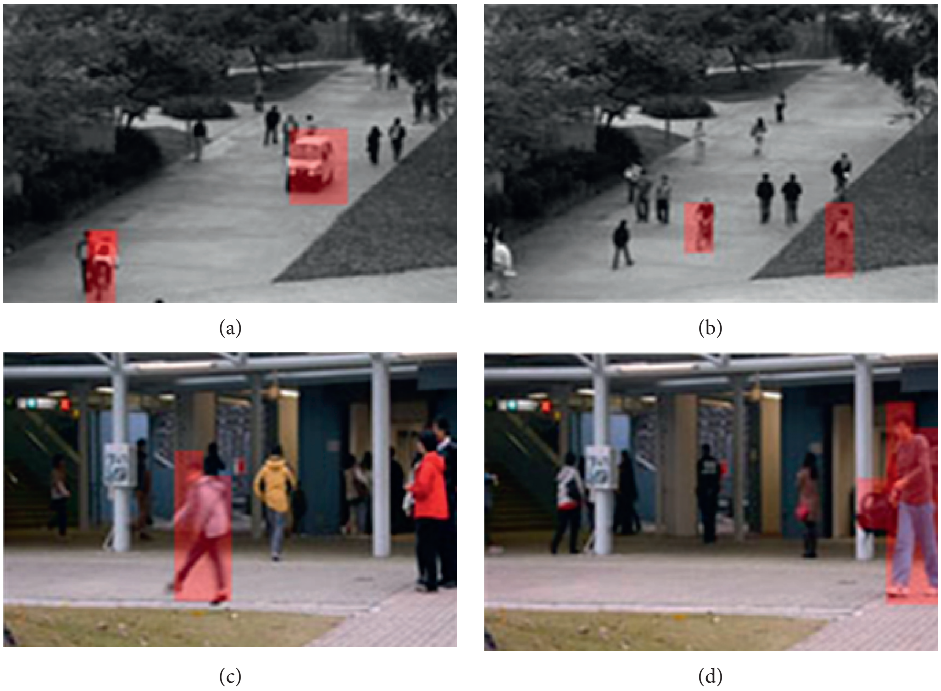

(a)
(b)
(c)
(d)

Figure 3: Examples of the detection results.

Table 2 shows the frame-level detection results on the Avenue dataset. On the Avenue dataset, only Method 2 and Method 4 are tested. And, Method 4 does not give the corresponding ROC curve. Compared with the other two methods, the proposed method achieves 82.1% results in frame-level AUC, which is higher than the other two methods by 1.3% and 3.8%, respectively. The results prove that the proposed method achieves high detection accuracy and good generalization on the Avenue dataset.

Figure 3 shows examples of partially correct detection results on two datasets. Among them, (a) and (b) are the test results of the USCD Ped1 dataset and (c) and (d) are the test results from the Avenue dataset. It can be observed from Figure 3 that the proposed method can detect different types of abnormal events, including bicycle, trolley, skateboard, and trolley. So, its performance for anomaly detection can be further validated.

Table 3 shows the comparison of detection speed between the proposed method and other one on the UCSD ped1 dataset. The results of the comparison methods come from their corresponding articles. The hardware environment of the whole experiment process is Intel Core i7-8700 k 3.7 GHz CPU, NVIDIA GeForce GTX 1070Ti (8 GB video memory) GPU and 16 GB RAM memory. The computing platform is *Python* 3.7 and Tensorflow 1.7. As can be seen from Table 3, the detection speed of the proposed method is 571 fps, which obviously surpasses the detection speed of other comparison methods.

Table 3: Running time comparison on the UCSD Ped1 dataset.

Method type	Computing platform	CPU (GHz)	GPU	RAM (GB)	Detection speed (fps)
Proposed	*Python* 3.7+Tensorflow1.7	3.7	NVIDIA GTX 1070Ti	16	571
Method 1	—	3.0	—	2.0	0.04
Method 2	MATLAB 2012	3.4	—	8.0	143.5
Method 3	MATLAB 2015	3.5	—	16	120
Method 4	MATLAB 2015	2.1	Nvidia quadro K4000	32	0.11

4. Conclusion

In this paper, a method of video anomaly detection and location based on VAE is proposed using an end-to-end deep learning framework. The method assumes that all normal samples conform to a Gaussian distribution. The probability value of the abnormal sample in the Gaussian distribution is relatively small. In the proposed method, the two steps of event representation and establishment of anomaly detection model are, respectively, converted into the hidden layer representation and Gaussian distribution constraint in the VAE. In addition, the two steps are jointly optimized to improve the accuracy and generalization ability of the method. The quantitative results in the two public datasets show that the proposed method has reached the current technological development level. The next step of the research will consider the realization of the proposed method on more complex datasets.

Data Availability

The datasets used in this paper are publicly available.

Conflicts of Interest

The authors author declares that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was sponsored in part by National Natural Science Foundation of China (61671309).

References

[1] H. Song, C. Sun, X. Wu, M. Chen, and Y. Jia, “Learning normal patterns via adversarial attention-based autoencoder for abnormal event detection in videos,” *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 2138–2148, 2020.

[2] S. Lee, H. G. Kim, and R. M. Ro, “BMAN: bidirectional multiscale Aggregation networks for abnormal event detection,” *IEEE Transactions on Image Processing*, vol. 29, pp. 2395–2408, 2019.

[3] O. Poppla and K. Wang, “Video-based abnormal human behavior recognition -a review,” *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, vol. 42, no. 6, pp. 865–878, 2012.

[4] A. Bera, S. Kim, and D. Mancoch, “Real-time anomaly detection using trajectory-level crowd behavior learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1289–1296, IEEE, Las Vegas, NV, USA, June 2016.

[5] O. Ye, J. Deng, Z. Yu, T. Liu, and L. Dong, “Abnormal event detection via feature expectation subgraph calibrating classification in video surveillance scenes,” *IEEE Access*, vol. 8, pp. 97564–97575, 2020.

[6] C. Lu, J. Shi, and J. Jia, “Abnormal event detection at 150 FPS in MATLAB,” in *Proceedings of IEEE International Conference on Computer Vision*, pp. 2720–2727, ICCV), December 2013, Sydney, Australia.

[7] Y. Cong, J. Yuan, and J. Liu, “Abnormal event detection in crowded scenes using sparse representation,” *Pattern Recognition*, vol. 46, no. 7, pp. 1851–1864, 2013.

[8] Y. Yuan, J. Fang, and Q. Wang, “Online anomaly detection in crowd scenes via structure analysis,” *IEEE Transactions on Cybernetics*, vol. 45, no. 3, pp. 562–575, 2015.

[9] W. Li, V. Mahadevan, and N. Vasconcelos, “Anomaly detection and localization in crowded scenes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 18–32, 2014.

[10] J. Kwon and K. M. Lee, “A unified framework for event summarization and rare event detection from multiple views,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1737–1750, 2015.

[11] H. Ren, W. Liu, S. I. Olsen, S. Escalera, and T. B. Moeslund, “Unsupervised behavior-specific dictionary learning for abnormal event detection,” in *Proceedings of the British Machine Vision Conference*, Swansea, Wales, September 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, IEEE, Las Vegas, NV, USA, June 2016.

[13] Z. Cai, N. Vasconcelos, and R.-C. N. N. Cascade, “Delving into high quality object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2348–2357, IEEE, Salt Lake City, UT, USA, June 2018.

[14] Y. Zhao, Y. Xiong, and D. Lin, “Recognize actions by disentangling components of dynamics,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2348–2357, IEEE, Salt Lake City, UT, USA, June 2018.

[15] D. Xu, Y. Yan, E. Ricci, and N. Sebe, “Detecting anomalous events in videos by learning deep representations of appearance and motion,” *Computer Vision and Image Understanding*, vol. 156, pp. 117–127, 2017.

[16] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, “A survey on deep learning for big data,” *Information Fusion*, vol. 42, pp. 146–157, 2018.

[17] M. Sabokrou, M. Fathy M, and M. Hoseini, “Video anomaly detection and localization based on the sparsity and reconstruction error of autoencoder,” *IET Electronic Letter*, vol. 52, no. 13, pp. 1122–1124, 2016.

[18] M. Hasan, J. Choi, J. Neumanny, A. K. Roy-Chowdhury, and L. S. Davis, “Learning temporal regularity in video sequence,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, IEEE, Las Vegas, NV, USA, June 2016.

[19] R. Xie, N. M. Jan, K. Hao, L. Chen, and B. Huang, "Supervised variational autoencoders for soft sensor modeling with missing data," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2820–2828, 2020.

[20] Y. Zerrouki, F. Harrou, N. Zerrouki, A. Dairi, and Y. Sun, "Desertification detection using an improved variational autoencoder-based approach through ETM-landsat satellite data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 202–213, 2021.

[21] Y. Wang, B. Dai, G. Hua, J. Aston, and D. Wipf, "Recurrent variational autoencoders for learning nonlinear generative models in the presence of outliers," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1615–1627, 2018.

[22] T. Zhao, F. Li, and P. Tian, "A deep-learning method for device activity detection in mMTC under imperfect CSI based on variational-autoencoder," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7981–7986, 2020.

[23] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proceedings of the International Conference on Learning Representations*, Banff, CA, USA, April 2014.

[24] P. Vincent, H. Larcochele, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, pp. 1096–1103, Helsinki, Finland, June 2008.

[25] M. Song, A. Montanari, and P.-M. Nguyen, "A mean field view of the landscape of two-layer neural networks," *Proceedings of the National Academy of Sciences*, vol. 115, no. 33, pp. E7665–E7671, 2018.

[26] T. Wang and H. Snoussi, "Detection of abnormal visual events via global optical flow orientation histogram," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 988–998, 2014.