Anomaly Detection in Surveillance Videos Using Spatio-Temporal Context Information

Roger Figueroa Quintero and Hernan D. Benitez-Restrepo
Pontificia Universidad Javeriana-Seccional Cali
Calle 18 No 118-250, Cali, Colombia
roger.figueroa@javerianacali.edu.co,hbenitez@javerianacali.edu.co

Abstract

Several computer vision algorithms have been proposed to detect anomalous activities (robberies, murders, vandalism, among others) in videos. According to the learning approach, they can be classified into probabilistic distribution modeling, sparse coding, and deep learning-based methods. The main drawbacks of these approaches are (i) extraction of low-level features that do not capture complex behaviors of instances on the scene, (ii) generation of features from irrelevant regions, (iii) overlooking of relationships among objects, and (iv) omission of long-term dependencies. To solve these issues, we propose a deep learning architecture that leverages the relationships among objects. It achieves this by using an attention mechanism and learning long-term dependencies using a multilayer recurrent neural network (multilayer LSTM). An AUC score of 0.749 on the UCF-Crime dataset confirms that the proposed algorithm competes effectively against several state-of-the-art approaches for anomaly detection in surveillance videos. It also explains the relationship between regions in the video frames and the anomaly detections.

1. Introduction

The videos captured by security cameras at different strategic locations in the cities are stored in a central monitoring station where a team of operators evaluates them for forensic analysis or for carrying out statistical reports on the city's security. Nonetheless, nowadays, the number of surveillance cameras in many cities worldwide has increased drastically, and manually assessing videos is a time-consuming and cumbersome task.

Several computer vision algorithms have been proposed to automate this task. Most of them often extract low-level features such as optical flow [2], dynamic texture (DT) [5], the mixture of dynamic textures (MDT) [13], cuboids [14] and convolutional network representation [20].

Most of these methods have been conceived to work in simple scenarios in which anomalous activities consist of a single type, such as abnormal pedestrian walkways, people groups escaping in panic, or moving in the wrong direction. In these conditions, algorithms work well because the anomalous activity is characterized by a drastic change in the scene motion patterns. Nonetheless, they do not perform well in complex street scenes where anomalous activity results from the interaction between the objects present in the scene (spatial context) and its temporal evolution (temporal context). [6, 13, 14, 16, 22]. Our approach introduces an algorithm that constructs the representation of the video segments based on the relationships of the objects on the scene and uses it to feed a recurrent neuronal network that learns the long-term dependencies. Thus, the resulting algorithm can detect not only anomalous activities characterized by the drastic change of motion patterns but also complex anomalous activities that result from long spatio-temporal interactions among objects in the scene, such as an armed robbery or a shooting. We present a spatial representation that captures the features of objects and their relationships on the scene based on Faster R-CNN [17], hierarchical clustering, the projection of each region of interest into C3D [21] feature map, and the extraction of a fixed vector feature by a RoIAlign layer [9]. In addition, we develop an attention model [1] that focuses on regions where an anomaly occurs. This model computes the convex combination of the feature representation extracted from different regions on the scene, giving higher weights to the features extracted from anomalous regions. Finally, we present a recurrent neural network-based architecture that detects anomalous activities at multiple temporal scales relying on the convex combination of the feature representation computed by the attention model.

2. Proposed Video Anomaly Detection Method

2.1. General architecture

We first divide the input video into small segments and extract the C3D features from different regions of the scene for each segment. Then, we employ an attention mechanism to weigh these features based on their importance, resulting in a fixed vector (attended vector). This vector is then fed into a multilayer LSTM, which captures the long-term context and predicts at each time step a set of temporal windows that enclose abnormal events. The complete architecture is depicted in Fig. 1.

2.2. Proposal regions

Most approaches for abnormal event detection rely on features extracted from complete frames. In contrast, our architecture detects abnormal events based on features extracted from the most important regions. We propose that the most important regions enclose groups of objects that possibly maintain a relationship in the scene, for example, a group of people chatting with each other. Given a set of objects in a scene, the number of partitions that could form grows according to the Bell sequence. However, not all partitions are plausible. To find plausible partitions, we resort to the rules for group formation presented in [18]. Here, the authors propose three rules that a clustering algorithm for group formation must hold: hierarchical coherence (HC), density invariance (DI), and transitivity. HC implies that groups are composed of individuals and subgroups in a recursive fashion. DI refers to obtaining similar clustering results despite crowd densities. Transitivity relaxes cluster formation, allowing two members to be part of the same group by means of a subgroup standing between them. Specifically, we used single-linkage. Given a set of bounding boxes, O_t, we cluster them by using single-linkage to create a hierarchical cluster H_t. Notice that each item in H_t is a cluster of bounding boxes. We then generate a new set P_t, by finding the minimum bounding box that encloses the regions of each item in H_t. We will refer to P_t as the set of "proposal regions."

2.3. Feature extraction

At each time step, the feature extraction process aims to obtain a fixed-length feature vector for each proposed region in P_t. To achieve this, we first discretize a video of length L into $T = L/\delta$ non-overlapping video segments, where δ=16 is the length of the video segment in frames. For each segment, we use C3D to extract the conv-5b feature maps Φ_t. We then project each proposal region onto the feature maps and feed the resulting tensor to the RoIAlign layer [9]. The output of the RoIAlign layer is a fixed-length feature vector representation. The set of feature vectors obtained following the aforementioned procedure is referred to as Z_t.

2.4. Multilayer LSTM and attention-mechanism

At each time step t, we extract the features from the proposal regions and employ an attention mechanism to combine them as follows:

$$\hat{\boldsymbol{z}}_t = \sum_{\boldsymbol{z}_{i,t} \in Z_t} \alpha_{i,t} \boldsymbol{z}_{i,t} \tag{1}$$

The combination's weights (see Eq. (4) and Eq. (5)) are updated for each time step t and can be thought of as the importance factor that the model assigns to each region. Next, we input the attended vector $\hat{\boldsymbol{z}}_t$ into a module consisting of three LSTM layers [7, 8], which produces a hidden state $\boldsymbol{h}_t$. Since $\boldsymbol{h}_t$ contains the accumulated temporal context up to time step t, we use it to feed a fully connected layer that predicts the probability of the occurrence of an anomalous event before time step t. Let us define a set of k temporal windows at time t as:

$$\rho_t = \{(t - b_j, t)\}_{j=1}^{k} \tag{2}$$

where the tuple $(t - b_j, t)$ indicates the start and end bounds of the j-th window. The probability with which the fully connected layer believes that an anomalous event occurred in the j-th window is given by the j-th entry of its output $\hat{\boldsymbol{y}}_t$:

$$\hat{\boldsymbol{y}}_t = \sigma_g \left(W_y \cdot \boldsymbol{h}_t \right) \tag{3}$$

where σ_g is the sigmoid activation functions and W_y is a learnable parameters.

We also utilize the hidden state of the multilayer LSTM to provide global context to the attention model. Specifically, given the hidden state of the previous time step, $\boldsymbol{h}_{t-1}$, and the set of feature vectors Z_t, we calculate the attention weights $\boldsymbol{\alpha}_t$ as follows:

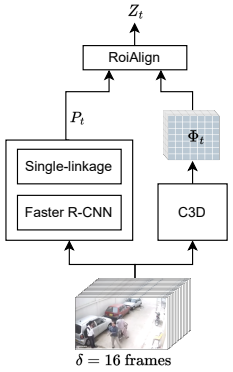
$$a_{i,t} = \boldsymbol{w}_a^T \tanh \left(W_{za} \boldsymbol{z}_{i,t} + W_{ha} \boldsymbol{h}_{t-1} \right) \tag{4}$$

$$\boldsymbol{\alpha}_t = \mathrm{softmax} \left(\boldsymbol{a}_t \right) \tag{5}$$

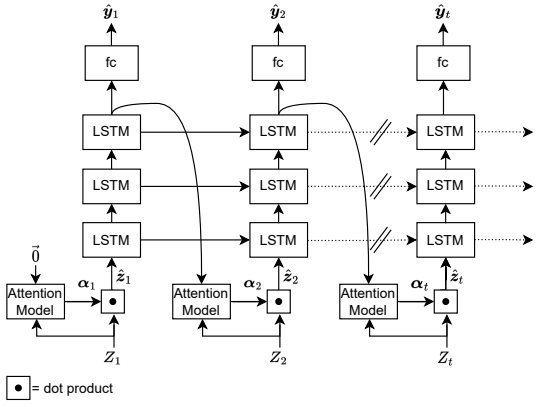
where $W_{za} \in R^{dc \times dz}$, $W_{ha} \in R^{dc \times dh}$ and $\boldsymbol{w}_a \in R^{dc}$ are parameters to be learnt.

2.5. Objective

During the training phase, we train our architecture to predict temporal windows that capture anomalous events. To achieve this, we follow the approach presented in [4], where the training target is derived from computing the temporal Intersection-over-Union (tIoU) between the ground truth of the training videos and the set of k proposals. Specifically, for each training video at each time step t, we



(a) Feature extraction.



(b) Attention model and multilayer LSTM.

Figure 1. General architecture of our Spatio-Temporal Anomaly Method (STAM). Given an input video, we divide it into short video segments consisting of 16 frames each. For each video segment, we detect a set of regions that enclose groups of objects in the scene (P_t). We then extract features for each region (Z_t) and weigh them using an attention model. The resulting attended vector ($\hat{\boldsymbol{z}}_t$) is fed into a multilayer LSTM, which retains the long-term context and predicts, at each time step, the probabilities ($\hat{\boldsymbol{y}}_t$) of an anomalous event has occurred within a set of time windows.

set the j-th entry of the target vector $\boldsymbol{y}_t$ to 1 if the tIoU between the ground truth and the j-th window (see Eq. (2)) is greater than 0.5, and 0 otherwise. The total loss for a training video is computed as follows:

$$\mathcal{L} = -\sum_{t=1}^{T}\sum_{j=1}^{k} y_t^j \log \hat{y}_t^j + \left(1 - y_t^j\right) \log\left(1 - \hat{y}_t^j\right) \quad (6)$$

where T is the number of segments in which was split the video.

3. Experiments

We evaluate our deep network architecture using the UCF-Crime Dataset [19] and compared its performance with other state-of-the-art methods. Specifically, we select three algorithms: (SBAD) [14] based on sparse coding, (MDI) [3] based on probabilistic methods, and (RWAD) [19] based on deep learning. As a performance metric, we compute the ROC and the AUC score following the 'Frame Level' methodology [13].

3.1. Training STAM

Because of the RAM limitations of our GPU, we do not use videos longer than 5 minutes. Therefore, starting from the initial training set, we derived a balanced data set consisting of 602 videos with anomalous activities and 602 videos with normal activities. We use this final dataset to train our architecture in supervised mode. As UCF-Crime does not provide ground truth annotations for training videos, we utilized the annotations provided by [11].

3.2. Implementation Details

The first step for training our architecture is to detect objects in the training videos. Generally, many available pretrained Faster R-CNN weights have been obtained by training on datasets with well-defined objects. However, surveillance videos are often of low quality. To address this issue, we trained Faster R-CNN on the MIO-TCD dataset [15] following the procedure described in [17]. We choose this dataset for two reasons. First, all its frames were captured by thousands of traffic surveillance cameras deployed all over Canada and the United States. Second, the MIO-TCD dataset contains 11 typical urban street objects, such as pedestrians, bicycles, motorcycles, cars, trucks, and buses. After training, we employ Faster R-CNN to detect objects every 16 frames and select the nine bounding boxes with the highest confidence score to construct the hierarchical clusters.

We implement the multilayer LSTM and attention model block in C++ using the Caffe library [10]. During training, we unroll the recurrent neural network for 562 time steps to cover the maximum time length of our dataset. We generate temporal proposals with 25 windows, that's sizes increase exponentially from 20 to 462 time steps. To address sequences with lengths less than 562, we pad them with zeros on the right. For model optimization, we adopt the Adam update rule [12] with a fixed learning rate of 0.05, $\beta_1 = 0.9$, and $\beta_1 = 0.999$. We train the recurrent block on an Nvidia Titan Xp GPU with a batch size of 64.

3.3. Comparison with the state-of-the-art

Fig. 2 shows the results obtained in the evaluation. According to the AUC score, RWAD obtains the best re-

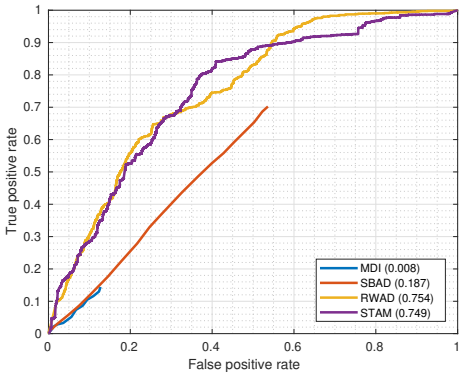


Figure 2. ROC curves of the evaluation following the frame-level methodology.

sult with 0.754, followed in order by the proposed method (STAM) with 0.749, SBAD with 0.187, and MDI with 0.008. The ROC curves for SBAD and MDI have few operating points, which indicates that these algorithms output similar anomaly scores for both positive and negative samples. One possible reason for the better performance of RWAD and our algorithm is their feature extraction phase. Whereas SBAD learns anomalous behavior at the pixel level and MDI learns from 2D convolutional features that only capture appearance, RWAD and STAM leverage C3D features that encode appearance and motion patterns.

Fig. 3 shows the results for a video in which the anomaly is an explosion. Only RWAD, STAM, and SBAD detect the explosion. After the explosion, only the score of RWAD decreases, while the scores of SBAD and STAM scores remain high. Despite the ground truth not indicating anomaly after the explosion, the effects remain until the end of the video. Therefore, the results of SBAD and STAM could be interpreted as correct. The lack of clear rules to define the starting and ending of an anomaly is precisely one of the problems of the UCF-Crime dataset. In contrast, MDI does not detect the explosion and it shows a false positive at the end of the video. Fig. 4 shows the results for a video in which the anomaly is an armed robbery. In this case, the scores of MDI and SBAD scores do not correlate with the scores from the ground truth, while the output scores of RWAD and STAM generate high anomaly scores for the anomalous frames.

4. Conclusions

We developed a DL-based architecture that leverages the spatio-temporal context in a scene to detect anomalous activities in surveillance videos. This architecture consists of an algorithm for region proposal generation, an algorithm for local feature extraction, an attention mechanism, and a stack of LSTMs. Although the proposed approach yields

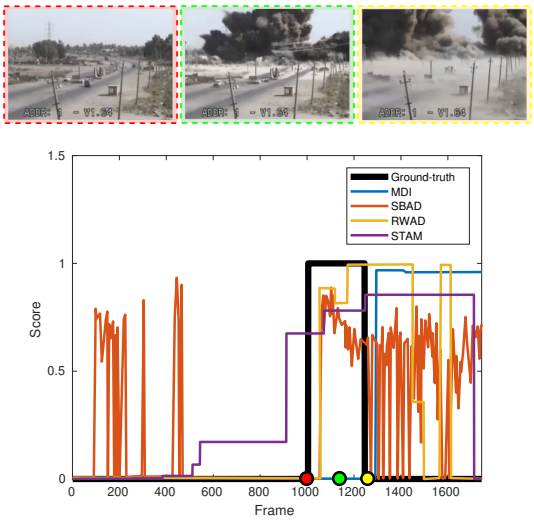


Figure 3. Output score for the "explosion" video category for the UCF-Crime dataset. The red, green, and yellow dots mark the numbers of the frames shown in the images above being enclosed with a rectangle of the same color.

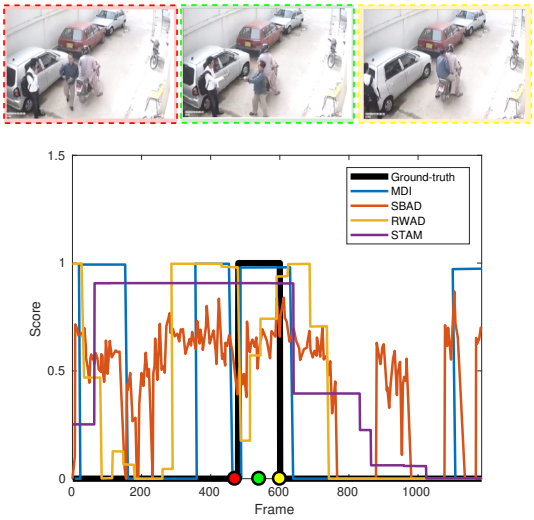


Figure 4. Output score for a "Robbery" category video for the UCF-Crime dataset. The red, green, and yellow dots mark the numbers of the frames shown in the images above being enclosed with a rectangle of the same color.

a lower AUC value on the UCF-Crime data set than competing state-of-the-art DL-based methods, it is important to note that our architecture relies on an attention model to provide explainability and discover the meaning between regions in the video frames and anomaly misdetections.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018. 1

[2] Ernesto L. Andrade, Scott Blunsden, and Robert B. Fisher. Modelling crowd scenes for event detection. In *Proceedings - International Conference on Pattern Recognition*, volume 1, pages 175–178, 2006. 1

[3] Björn Barz, Erik Rodner, Yanira Guanche Garcia, and Joachim Denzler. Detecting regions of maximal divergence for spatio-temporal anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 41(5):1088–1101, 2018. 3

[4] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. SST: Single-stream temporal action proposals. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua:6373–6382, 2017. 2

[5] Antoni B Chan, Student Member, and Nuno Vasconcelos. Modeling , Clustering , and Segmenting Video with Mixtures of Dynamic Textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):909–926, 2008. 1

[6] N. P. Cuntoor, B. Yegnanarayana, and R. Chellappa. Activity modeling using event probability sequences. *IEEE Transactions on Image Processing*, 17(4):594–607, 2008. 1

[7] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005. 2

[8] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232, 2016. 2

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):386–397, 2020. 1, 2

[10] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 3

[11] Hennie Kim. action recognition study. https://github.com/henniekim/action_recognition_study, 2018. 3

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[13] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):18–32, 2014. 1, 3

[14] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 FPS in MATLAB. *Proceedings of the IEEE International Conference on Computer Vision*, pages 2720–2727, 2013. 1, 3

[15] Zhiming Luo, Frederic Branchaud-Charron, Carl Lemaire, Janusz Konrad, Shaozi Li, Akshaya Mishra, Andrew Achkar, Justin Eichel, and Pierre-Marc Jodoin. Mio-tcd: A new benchmark dataset for vehicle classification and localization. *IEEE Transactions on Image Processing*, 27(10):5129–5141, 2018. 3

[16] O P Popoola and Kejun Wang. Video-Based Abnormal Human Behavior Recognition: A Review. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(6):865–878, 2012. 1

[17] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. 1, 3

[18] Francesco Solera, Simone Calderara, and Rita Cucchiara. Socially constrained structural learning for groups detection in crowd. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):995–1008, 2015. 2

[19] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world Anomaly Detection in Surveillance Videos. *arXiv preprint:1801.04264*, 2018. 3

[20] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision*, 2015 Inter:4489–4497, 2015. 1

[21] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1

[22] Tao Xiang and Shaogang Gong. Video Behavior Profiling for Anomaly Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):893–908, 2008. 1