Real-world Anomaly Detection in Surveillance Videos

Waqas Sultani[1]
[1]Department of Computer Science
Information Technology University, Pakistan

Chen Chen[2], Mubarak Shah[2]
[2]Center for Research in Computer Vision
University of Central Florida, Orlando, FL,USA

waqas5163@gmail.com, chenchen870713@gmail.com, shah@crcv.ucf.edu

Abstract

Surveillance videos are able to capture a variety of realistic anomalies. In this paper, we propose to learn anomalies by exploiting both normal and anomalous videos. To avoid annotating the anomalous segments or clips in training videos, which is very time consuming, we propose to learn anomaly through the deep multiple instance ranking framework by leveraging weakly labeled training videos, i.e. *the training labels (anomalous or normal) are at video-level instead of clip-level. In our approach, we consider normal and anomalous videos as bags and video segments as instances in multiple instance learning (MIL), and automatically learn a deep anomaly ranking model that predicts high anomaly scores for anomalous video segments. Furthermore, we introduce sparsity and temporal smoothness constraints in the ranking loss function to better localize anomaly during training.*

We also introduce a new large-scale first of its kind dataset of 128 hours of videos. It consists of 1900 long and untrimmed real-world surveillance videos, with 13 realistic anomalies such as fighting, road accident, burglary, robbery, etc. as well as normal activities. This dataset can be used for two tasks. First, general anomaly detection considering all anomalies in one group and all normal activities in another group. Second, for recognizing each of 13 anomalous activities. Our experimental results show that our MIL method for anomaly detection achieves significant improvement on anomaly detection performance as compared to the state-of-the-art approaches. We provide the results of several recent deep learning baselines on anomalous activity recognition. The low recognition performance of these baselines reveals that our dataset is very challenging and opens more opportunities for future work. The dataset is available at: http://crcv.ucf.edu/projects/real-world/

1. Introduction

Surveillance cameras are increasingly being used in public places *e.g.* streets, intersections, banks, shopping malls, etc. to increase public safety. However, the monitoring capability of law enforcement agencies has not kept pace. The result is that there is a glaring deficiency in the utilization of surveillance cameras and an unworkable ratio of cameras to human monitors. One critical task in video surveillance is detecting anomalous events such as traffic accidents, crimes or illegal activities. Generally, anomalous events rarely occur as compared to normal activities. Therefore, to alleviate the waste of labor and time, developing intelligent computer vision algorithms for automatic video anomaly detection is a pressing need. The goal of a practical anomaly detection system is to timely signal an activity that deviates normal patterns and identify the time window of the occurring anomaly. Therefore, anomaly detection can be considered as coarse level video understanding, which filters out anomalies from normal patterns. Once an anomaly is detected, it can further be categorized into one of the specific activities using classification techniques.

A small step towards addressing anomaly detection is to develop algorithms to detect a specific anomalous event, for example violence detector [30] and traffic accident detector [23, 35]. However, it is obvious that such solutions cannot be generalized to detect other anomalous events, therefore they render a limited use in practice.

Real-world anomalous events are complicated and diverse. It is difficult to list all of the possible anomalous events. Therefore, it is desirable that the anomaly detection algorithm does not rely on any prior information about the events. In other words, anomaly detection should be done with minimum supervision. Sparse-coding based approaches [28, 42] are considered as representative methods that achieve state-of-the-art anomaly detection results. These methods assume that only a small initial portion of a video contains normal events, and therefore the initial portion is used to build the normal event dictionary. Then, the main idea for anomaly detection is that anomalous events are not accurately reconstructable from the normal event dictionary. However, since the environment captured by

surveillance cameras can change drastically over the time (*e.g*. at different times of a day), these approaches produce high false alarm rates for different normal behaviors.

Motivation and contributions. Although the above-mentioned approaches are appealing, they are based on the assumption that any pattern that deviates from the learned normal patterns would be considered as an anomaly. However, this assumption may not hold true because *it is very difficult or impossible to define a normal event which takes all possible normal patterns/behaviors into account* [9]. More importantly, the boundary between normal and anomalous behaviors is often ambiguous. In addition, under realistic conditions, the same behavior could be a normal or an anomalous behavior under different conditions. In this paper, we propose an anomaly detection algorithm using weakly labeled training videos. That is we only know the video-level labels, *i.e*. *a video is normal or contains anomaly somewhere, but we do not know where.* This is intriguing because we can easily annotate a large number of videos by only assigning video-level labels. To formulate a weakly-supervised learning approach, we resort to multiple instance learning (MIL) [12, 4]. Specifically, we propose to learn anomaly through a deep MIL framework by treating normal and anomalous surveillance videos as bags and short segments/clips of each video as instances in a bag. Based on training videos, we automatically learn an anomaly *ranking model* that predicts high anomaly scores for anomalous segments in a video. During testing, a long-untrimmed video is divided into segments and fed into our deep network which assigns anomaly score for each video segment such that an anomaly can be detected. In summary, this paper makes the following contributions.

• We propose a MIL solution to anomaly detection by leveraging only weakly labeled training videos. We propose a MIL ranking loss with sparsity and smoothness constraints for a deep learning network to learn anomaly scores for video segments.

• We introduce a large-scale video anomaly detection dataset consisting of 1900 real-world surveillance videos of 13 different anomalous events and normal activities captured by surveillance cameras. It is by far the largest dataset with more than 25 times videos than existing largest anomaly dataset and has a total of 128 hours of videos.

• Experimental results on our new dataset show that our proposed method achieves superior performance as compared to the state-of-the-art anomaly detection approaches.

• Our dataset also serves a challenging benchmark for activity recognition on *untrimmed* videos, due to the complexity of activities and large intra-class variations. We provide results of baseline methods, C3D [37] and TCNN [21], on recognizing 13 different anomalous activities.

2. Related Work

Anomaly detection. Anomaly detection is one of the most challenging and long standing problems in computer vision [40, 39, 7, 10, 5, 20, 43, 27, 26, 28, 42, 18, 26]. For video surveillance applications, there are several attempts to detect violence or aggression [15, 25, 11, 30] in videos. Datta *et al*. proposed to detect human violence by exploiting motion and limbs orientation of people. Kooij *et al*. [25] employed video and audio data to detect aggressive actions in surveillance videos. Gao *et al*. proposed violent flow descriptors to detect violence in crowd videos. More recently, Mohammadi *et al*. [30] proposed a new behavior heuristic based approach to classify violent and non-violent videos.

Beyond violent and non-violent patterns discrimination, authors in [39, 7] proposed to use tracking to model the normal motion of people and detect deviation from that normal motion as an anomaly. Due to difficulties in obtaining reliable tracks, several approaches avoid tracking and learn global motion patterns through histogram-based methods [10], topic modeling [20], motion patterns [32], social force models [29], mixtures of dynamic textures model [27], Hidden Markov Model (HMM) on local spatio-temporal volumes [26], and context-driven method [43]. Given the training videos of normal behaviors, these approaches learn distributions of normal motion patterns and detect low probable patterns as anomalies.

Following the success of sparse representation and dictionary learning approaches in several computer vision problems, researchers in [28, 42] used sparse representation to learn the dictionary of normal behaviors. During testing, the patterns which have large reconstruction errors are considered as anomalous behaviors. Due to successful demonstration of deep learning for image classification, several approaches have been proposed for video action classification [24, 37]. However, obtaining annotations for training is difficult and laborious, specifically for videos.

Recently, [18, 40] used deep learning based autoencoders to learn the model of normal behaviors and employed reconstruction loss to detect anomalies. Our approach not only considers normal behaviors but also anomalous behaviors for anomaly detection, using only weakly labeled training data.

Ranking. Learning to rank is an active research area in machine learning. These approaches mainly focused on improving relative scores of the items instead of individual scores. Joachims *et al*. [22] presented rank-SVM to improve retrieval quality of search engines. Bergeron *et al*. [8] proposed an algorithm for solving multiple instance ranking problems using successive linear programming and demonstrated its application in hydrogen abstraction problem in computational chemistry. Recently, deep ranking networks have been used in several computer vision applications and have shown state-of-the-art performances. They

have been used for feature learning [38], highlight detection [41], Graphics Interchange Format (GIF) generation [17], face detection and verification [33], person re-identification [13], place recognition [6], metric learning and image retrieval [16]. All deep ranking methods require a vast amount of annotations of positive and negative samples.

In contrast to the existing methods, we formulate anomaly detection as a regression problem (we call it regression since we map feature vector to an anomaly score (0-1)) in the ranking framework by utilizing normal and anomalous data. To alleviate the difficulty of obtaining precise segment-level labels (*i.e*. temporal annotations of the anomalous parts in videos) for training, we leverage multiple instance learning which relies on weakly labeled data (*i.e*. video-level labels – normal or abnormal, which are much easier to obtain than temporal annotations) to learn the anomaly model and detect video segment level anomaly during testing.

3. Proposed Anomaly Detection Method

The proposed approach (summarized in Figure 1) begins with dividing surveillance videos into a fixed number of segments during training. These segments make instances in a bag. Using both positive (anomalous) and negative (normal) bags, we train the anomaly detection model using the proposed deep MIL ranking loss.

3.1. Multiple Instance Learning

In standard supervised classification problems using support vector machine, the labels of all positive and negative examples are available and the classifier is learned using the following optimization function:

$$\min_{\mathbf{w}} \ \frac{1}{k}\sum_{i=1}^{k} \overbrace{max(0, 1 - y_i(\mathbf{w}.\phi(x) - b))}^{①} + \frac{1}{2}\left\|\mathbf{w}\right\|^2, \quad (1)$$

where ① is the hinge loss, y_i represents the label of each example, $\phi(x)$ denotes feature representation of an image patch or a video segment, b is a bias, k is the total number of training examples and $\mathbf{w}$ is the classifier to be learned. To learn a robust classifier, accurate annotations of positive and negative examples are needed. In the context of supervised anomaly detection, a classifier needs temporal annotations of each segment in videos. However, obtaining temporal annotations for videos is time consuming and laborious.

MIL relaxes the assumption of having these accurate temporal annotations. In MIL, precise temporal locations of anomalous events in videos are unknown. Instead, only video-level labels indicating the presence of an anomaly in the *whole* video is needed. A video containing anomalies is labeled as positive and a video without any anomaly is labeled as negative. Then, we represent a positive video as a positive bag $\mathcal{B}_a$, where different temporal segments make individual instances in the bag, $(p^1, p^2, \ldots, p^m)$, where m is the number of instances in the bag. We assume that at least one of these instances contains the anomaly. Similarly, the negative video is denoted by a negative bag, $\mathcal{B}_n$, where temporal segments in this bag form negative instances $(n^1, n^2, \ldots, n^m)$. In the negative bag, none of the instances contain an anomaly. Since the exact information (*i.e*. instance-level label) of the positive instances is unknown, one can optimize the objective function with respect to the maximum scored instance in each bag [4]:

$$\min_{\mathbf{w}} \frac{1}{z}\sum_{j=1}^{z} max(0, 1 - Y_{\mathcal{B}_j}(\max_{i \in \mathcal{B}_j}(\mathbf{w}.\phi(x_i)) - b)) + \frac{1}{2}\left\|\mathbf{w}\right\|^2, \quad (2)$$

where $Y_{\mathcal{B}_j}$ denotes bag-level label, z is the total number of bags, and all the other variables are the same as in Eq. 1.

3.2. Deep MIL Ranking Model

Anomalous behavior is difficult to define accurately [9], since it is quite subjective and can vary largely from person to person. Further, it is not obvious how to assign 1/0 labels to anomalies. Moreover, due to the unavailability of sufficient examples of anomaly, anomaly detection is usually treated as low likelihood pattern detection instead of classification problem [10, 5, 20, 26, 28, 42, 18, 26].

In our proposed approach, we pose anomaly detection as a regression problem. We want the anomalous video segments to have higher anomaly scores than the normal segments. The straightforward approach would be to use a ranking loss which encourages high scores for anomalous video segments as compared to normal segments, such as:

$$f(\mathcal{V}_a) > f(\mathcal{V}_n), \quad (3)$$

where $\mathcal{V}_a$ and $\mathcal{V}_n$ represent anomalous and normal video segments, $f(\mathcal{V}_a)$ and $f(\mathcal{V}_n)$ represent the corresponding predicted anomaly scores ranging from 0 to 1, respectively. The above ranking function should work well if the segment-level annotations are known during training.

However, in the absence of video segment level annotations, it is not possible to use Eq. 3. Instead, we propose the following multiple instance ranking objective function:

$$\max_{i \in \mathcal{B}_a} f(\mathcal{V}_a^i) > \max_{i \in \mathcal{B}_n} f(\mathcal{V}_n^i), \quad (4)$$

where *max* is taken over all video segments in each bag. Instead of enforcing ranking on every instance of the bag, we enforce ranking only on the two instances having the highest anomaly score respectively in the positive and negative bags. The segment corresponding to the highest anomaly score in the positive bag is most likely to be the true positive instance (anomalous segment). The segment corresponding to the highest anomaly score in the negative bag is the one looks most similar to an anomalous segment but actually is

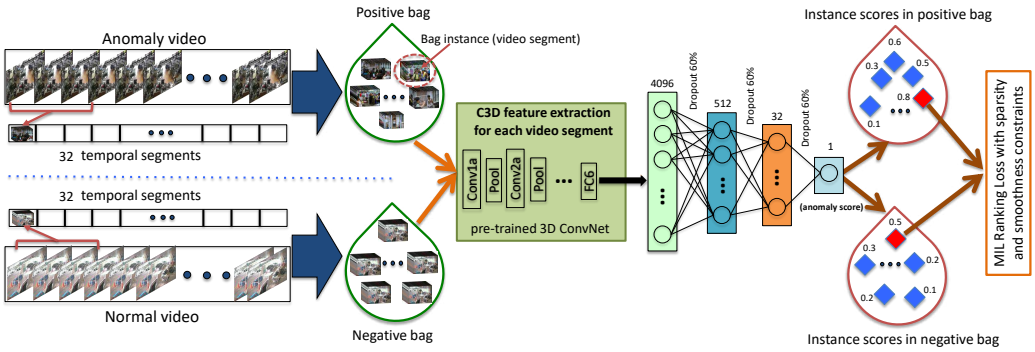

Figure 1. The flow diagram of the proposed anomaly detection approach. Given the positive (containing anomaly somewhere) and negative (containing no anomaly) videos, we divide each of them into multiple temporal video segments. Then, each video is represented as a bag and each temporal segment represents an instance in the bag. After extracting C3D features [37] for video segments, we train a fully connected neural network by utilizing a novel ranking loss function which computes the ranking loss between the highest scored instances (shown in red) in the positive bag and the negative bag.

a normal instance. This negative instance is considered as a hard instance which may generate a false alarm in anomaly detection. By using Eq. 4, we want to push the positive instances and negative instances far apart in terms of anomaly score. Our ranking loss in the hinge-loss formulation is therefore given as follows:

$$l(\mathcal{B}_a, \mathcal{B}_n) = \max(0, 1 - \max_{i \in \mathcal{B}_a} f(\mathcal{V}_a^i) + \max_{i \in \mathcal{B}_n} f(\mathcal{V}_n^i)). \quad (5)$$

One limitation of the above loss is that it ignores the underlying temporal structure of the anomalous video. First, in real-world scenarios, anomaly often occurs only for a short time. In this case, the scores of the instances (segments) in the anomalous bag should be sparse, indicating only a few segments may contain the anomaly. Second, since the video is a sequence of segments, the anomaly score should vary smoothly between video segments. Therefore, we enforce temporal smoothness between anomaly scores of temporally adjacent video segments by minimizing the difference of scores for adjacent video segments. By incorporating the sparsity and smoothness constraints on the instance scores, the loss function becomes

$$l(\mathcal{B}_a, \mathcal{B}_n) = \max(0, 1 - \max_{i \in \mathcal{B}_a} f(\mathcal{V}_a^i) + \max_{i \in \mathcal{B}_n} f(\mathcal{V}_n^i)) + \lambda_1 \overbrace{\sum_i^{(n-1)} (f(\mathcal{V}_a^i) - f(\mathcal{V}_a^{i+1}))^2}^{①} + \lambda_2 \overbrace{\sum_i^{n} f(\mathcal{V}_a^i)}^{②}, \quad (6)$$

where ① indicates the temporal smoothness term and ② represents the sparsity term. In this MIL ranking loss, the error is back-propagated from the maximum scored video segments in both positive and negative bags. By training on a large number of positive and negative bags, we expect that the network will learn a generalized model to predict high scores for anomalous segments in positive bags (see Figure 8). Finally, our complete objective function is given by

$$\mathcal{L}(\mathcal{W}) = l(\mathcal{B}_a, \mathcal{B}_n) + \lambda_3 \|\mathcal{W}\|_F, \quad (7)$$

where $\mathcal{W}$ represents model weights.

Bags Formations. We divide each video into the equal number of non-overlapping temporal segments and use these video segments as bag instances. Given each video segment, we extract the 3D convolution features [37]. We use this feature representation due to its computational efficiency and the evident capability of capturing appearance and motion dynamics in video action recognition.

4. Dataset

4.1. Previous datasets

We briefly review the existing video anomaly detection datasets in this section. The **UMN** dataset [2] consists of five different staged videos, where people walk around and after some time start running in different directions. The anomaly is characterized by only running action. **UCSD Ped1** and **Ped2** datasets [27] contain 70 and 28 surveillance videos, respectively. Those videos are captured at only one location. The anomalies in the videos are simple and do not reflect realistic anomalies in video surveillance, *e.g.* people walking across a walkway, non pedestrian entities (skater, biker and wheelchair) in the walkways. **Avenue** dataset [28] consists of 37 videos. Although it contains more anomalies, they are staged and captured at one location. Similar to [27], videos in this dataset are short and some of the anomalies are unrealistic (*e.g.* throwing paper). **Subway Exit** and **Subway Entrance** datasets [3] contain one long surveillance video each. The two videos capture simple anomalies such as walking in the wrong direction and skipping payment. **BOSS** [1] dataset is collected from a surveillance camera mounted in a train. It contains anomalies such as harassment, person with a disease, panic situation, as well as

normal videos. All anomalies are performed by actors. **Abnormal Crowd** [31] introduced a crowd anomaly dataset which contains 31 videos with crowded scenes only. Overall, the previous datasets for video anomaly detection are small in terms of the number of videos or the length of the video. Variations in abnormalities are also limited. In addition, some anomalies are not realistic.

4.2. Our dataset

Due to the limitations of previous datasets, we construct a new large-scale dataset to evaluate our method. It consists of long *untrimmed surveillance videos* which cover 13 real-world anomalies, including *Abuse, Arrest, Arson, Assault, Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting,* and *Vandalism*. These anomalies are selected because they have a significant impact on public safety. We compare our dataset with previous anomaly detection datasets in Table 1.

Video collection. To ensure the quality of our dataset, we train ten annotators (having different levels of computer vision expertise) to collect the dataset. We search videos on YouTube and LiveLeak [1] using text search queries (with slight variations *e.g*. "car crash", "road accident") of each anomaly. In order to retrieve as many videos as possible, we also use text queries in different languages (*e.g*. French, Russian, Chinese, etc.) for each anomaly, thanks to Google translator. We remove videos which fall into any of the following conditions: manually edited, prank videos, not captured by CCTV cameras, taking from news, captured using a hand-held camera, and containing compilation. We also discard videos in which the anomaly is not clear. With the above video pruning constraints, 950 unedited real-world surveillance videos with clear anomalies are collected. Using the same constraints, 950 normal videos are gathered, leading to a total of 1900 videos in our dataset. In Figure 2, we show four frames of an example video from each anomaly.

Annotation. For our anomaly detection method, only video-level labels are required for training. However, in order to evaluate its performance on testing videos, we need to know the temporal annotations, *i.e*. the start and ending frames of the anomalous event in each testing anomalous video. To this end, we assign the same videos to multiple annotators to label the temporal extent of each anomaly. The final temporal annotations are obtained by averaging annotations of different annotators. The complete dataset is finalized after intense efforts of several months.

Training and testing sets. We divide our dataset into two parts: the training set consisting of 800 normal and 810 anomalous videos (details shown in Table 2) and the testing set including the remaining 150 normal and 140 anomalous videos. Both training and testing sets contain all 13 anomalies at various temporal locations in the videos. Furthermore, some of the videos have multiple anomalies. The distribution of the training videos in terms of length (in minute) is shown in Figures 3. The number of frames and percentage of anomaly in each testing video are presented in Figures 4 and 5, respectively.

[1] https://www.youtube.com/ , https://www.liveleak.com/

5. Experiments

5.1. Implementation Details

We extract visual features from the fully connected (FC) layer FC6 of the C3D network [37]. Before computing features, we re-size each video frame to 240 $\times$ 320 pixels and fix the frame rate to 30 fps. We compute C3D features for every 16-frame video clip followed by l_2 normalization. To obtain features for a video segment, we take the average of all 16-frame clip features within that segment. We input these features (4096D) to a 3-layer FC neural network. The first FC layer has 512 units followed by 32 units and 1 unit FC layers. 60% dropout regularization [34] is used between FC layers. We use ReLU [19] activation and Sigmoid activation for the first and the last FC layers respectively, and employ Adagrad [14] optimizer with the initial learning rate of 0.001. The parameters of sparsity and smoothness constraints in the MIL ranking loss are set to λ_1=$\lambda_2 = 8 \times 10^{-5}$ and λ_3 = 0.01 for the best performance.

We divide each video into 32 non-overlapping segments and consider each video segment as an instance of the bag. The number of segments (32) is empirically set. We also experimented with multi-scale overlapping temporal segments but it does not affect detection accuracy. We randomly select 30 positive and 30 negative bags as a mini-batch. We compute gradients by reverse mode automatic differentiation on computation graph using Theano [36]. Then we compute loss as shown in Eq. 6 and Eq. 7 and back-propagate the loss for the whole batch.

Evaluation Metric. Following previous works on anomaly detection [27], we use frame based receiver operating characteristic (ROC) curve and corresponding area under the curve (AUC) to evaluate the performance of our method. We do not use equal error rate (EER) [27], as it does not measure anomaly correctly, specifically if only a small portion of a long video contains anomalous behavior.

5.2. Comparison with the State-of-the-art

We compare our method with two state-of-the-art approaches for anomaly detection. Lu *et al*. [28] proposed **dictionary based approach** to learn the normal behaviors and used reconstruction errors to detect anomalies. Following their code, we extract 7000 cuboids from each of the normal training video and compute gradient based features in each volume. After reducing the feature dimension us-

	# of videos	Average # of frames	Dataset length	Example anomalies
UCSD Ped1 [27]	70	201	5 min	Bikers, small carts, walking across walkways
UCSD Ped2 [27]	28	163	5 min	Bikers, small carts, walking across walkways
Subway Entrance [3]	1	121,749	1.5 hours	Wrong direction, No payment
Subwa Exit [3]	1	64,901	1.5 hours	Wrong direction, No payment
Avenue [28]	37	839	30 min	Run, throw, new object
UMN [2]	5	1290	5 min	Run
BOSS [1]	12	4052	27 min	Harass, disease, panic
Abnormal Crowd [31]	31	1408	24 min	Panic, fight, congestion, obstacle, neutral
Ours	**1900**	**7247**	**128 hours**	**Abuse, arrest, arson, assault, accident, burglary, fighting, robbery**

Table 1. A comparison of anomaly datasets. Our dataset contains larger number of longer surveillance videos with more realistic anomalies.

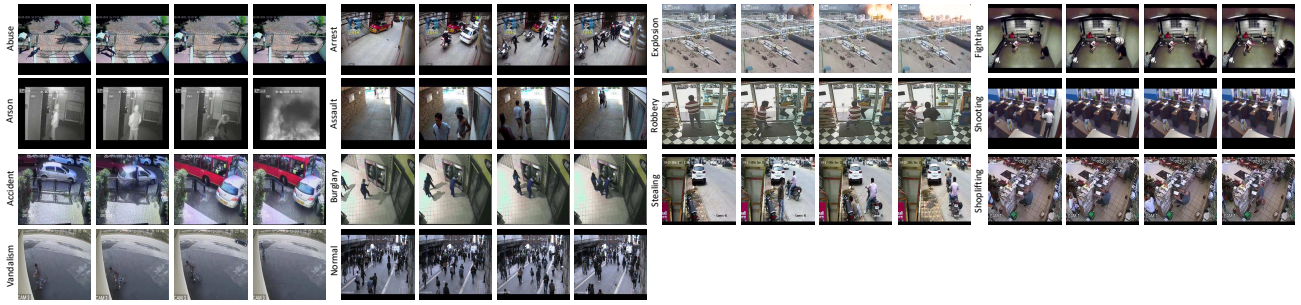

Figure 2. Examples of different anomalies in our dataset.

Anomaly	**# of videos**
Abuse	50 (48)
Arrest	50 (45)
Arson	50 (41)
Assault	50 (47)
Burglary	100 (87)
Explosion	50 (29)
Fighting	50 (45)
Road Accidents	150 (127)
Robbery	150 (145)
Shooting	50 (27)
Shoplifting	50 (29)
Stealing	100 (95)
Vandalism	50 (45)
Normal events	950 (800)

Table 2. Total number of videos of each anomaly in our dataset. Numbers in brackets represent the number of videos in the training set.

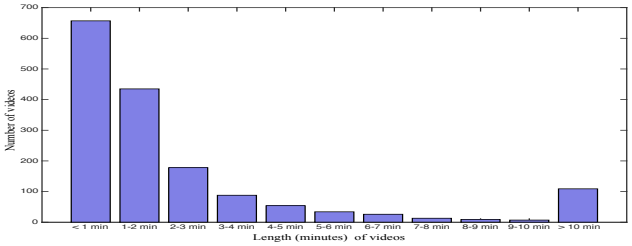

Figure 3. Distribution of videos according to length (minutes) in the training set.

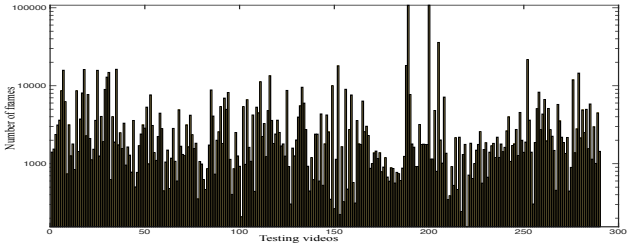

Figure 4. Distribution of video frames in the testing set.

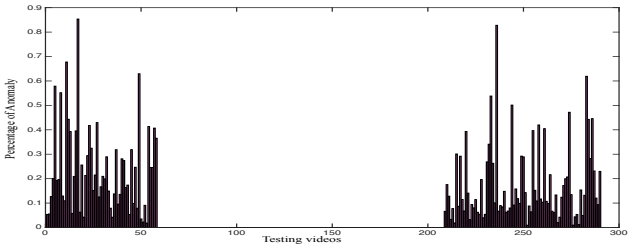

Figure 5. Percentage of anomaly in each video of the testing set. Normal videos (59 to 208) do not contain any anomaly.

ing PCA, we learn the dictionary using sparse representation. Hasan *et al*. [18] proposed a fully convolutional feed-forward **deep auto-encoder based approach** to learn local features and classifier. Using their implementation, we train the network on normal videos using the temporal window of 40 frames. Similar to [28], reconstruction error is used to measure anomaly. We keep the model training setting of this method similar to our proposed approach, *i.e*. 32 video segments in each bag with features computed using C3D. In addition, we also use a **binary SVM classifier** as a baseline method. Specifically, we treat all anomalous videos as one class and normal videos as another class. C3D features are computed for each video, and a binary classifier is trained with linear kernel. For testing, this classifier provides the

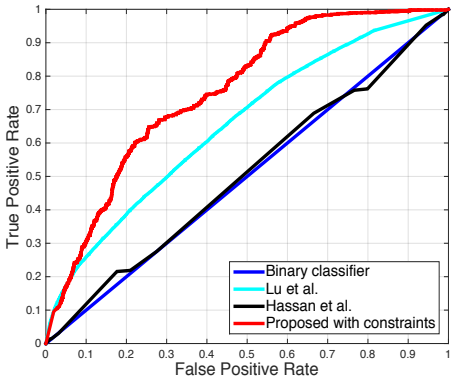

Figure 6. ROC comparison of binary classifier (blue), Lu *et al*. [28] (cyan), Hasan *et al*. [18] (black), proposed method without constraints (magenta) and with constraints (red).

probability of each video clip to be anomalous.

The quantitative comparisons in terms of ROC and AUC are shown in Figure 6 and Table 3. We also compare the results of our approach with and without smoothness and sparsity constraints. The results show that our approach significantly outperforms the existing methods. Particularly, our method achieves much higher true positive rates than other methods under low false positive rates *e.g*. 0.1-0.3.

The binary classifier results demonstrate that traditional action recognition approaches cannot be used for anomaly detection in real-world surveillance videos. This is because our dataset contains long untrimmed videos where anomaly mostly occurs for a short period of time. Therefore, the features extracted from these untrimmed training videos are not discriminative enough for the anomalous events. In the experiments, binary classifier produces very low anomaly scores for almost all testing videos. Dictionary learnt by [28] is not robust enough to discriminate between normal and anomalous pattern. In addition to producing the low reconstruction error for normal portion of the videos, it also produces low reconstruction error for anomalous part. Hasan *et al*. [18] learns normal patterns quite well. However, it tends to produce high anomaly scores even for new normal patterns. Our method performing significantly better than [18] demonstrates its effectiveness.

In Figure 7, we present qualitative results of our approach on eight videos. (a)-(d) show four videos with anomalous events. Our method provides successful and timely detection of those anomalies by generating high anomaly scores for the anomalous frames. (e) and (f) are two normal videos. Our method produces low anomaly scores (close to 0) through out the entire video, yielding zero false alarm for the two normal videos. We also illustrate two failure cases in (g) and (h). Specifically, (g) is an anomalous video containing a burglary event (person entering an office through a window). Our method fails to detect the anomalous part because of the darkness of the scene (a night video). Also, it generates false alarms mainly due to occlusions by flying insects in front of camera. In (h), our method produces false alarms due to sudden people gathering (watching a relay race in street). In other words, it fails to identify the normal group activity.

Method	AUC
Binary classifier	50.0
Hasan *et al*. [18]	50.6
Lu *et al*. [28]	65.51
Proposed w/o constraints	74.44
Proposed w constraints	**75.41**

Table 3. AUC comparison of various approaches on our dataset.

Method	[18]	[28]	**Proposed**
False alarm rate	27.2	3.1	**1.9**

Table 4. False alarm rate comparison on normal testing videos.

5.3. Analysis of the Proposed Method

Model training. The underlying assumption of the proposed approach is that given a lot of positive and negative videos with video-level labels, the network can automatically learn to predict the location of the anomaly in the video. To achieve this goal, the network should learn to produce high scores for anomalous video segments during training iterations. Figure 8 shows the evolution of anomaly score for a training anomalous example over the iterations. At 1,000 iterations, the network predicts high scores for both anomalous and normal video segments. After 3,000 iterations, the network starts to produce low scores for normal segments and keep high scores of anomalous segments. As the number of iterations increases and the network sees more videos, it automatically learns to precisely localize anomaly. Note that although we do not use any segment level annotations, the network is able to predict the temporal location of an anomaly in terms of anomaly scores.

False alarm rate. In real-world setting, a major part of a surveillance video is normal. A robust anomaly detection method should have low false alarm rates on normal videos. Therefore, we evaluate the performance of our approach and other methods on normal videos only. Table 4 lists the false alarm rates of different approaches at 50% threshold. Our approach has a much lower false alarm rate than other methods, indicating a more robust anomaly detection system in practice. This validates that using both anomalous and normal videos for training helps our deep MIL ranking model to learn more general normal patterns.

5.4. Anomalous Activity Recognition Experiments

Our dataset can be used as an anomalous activity recognition benchmark, since we have event labels for the anoma-

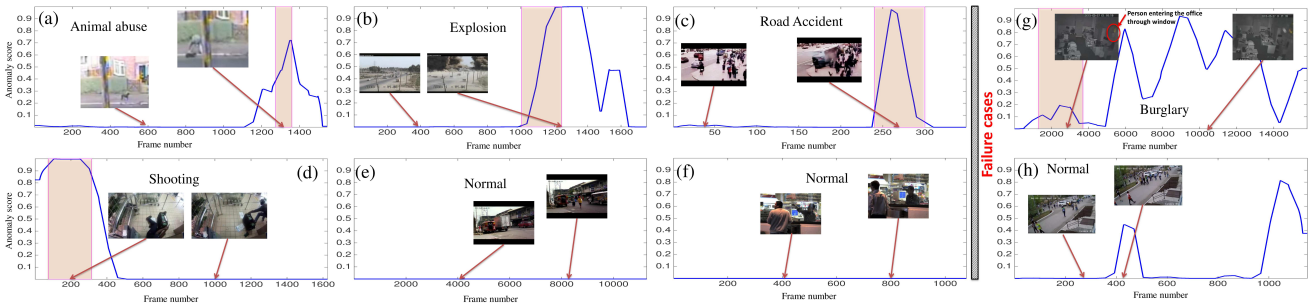

Figure 7. Qualitative results of our method on testing videos. Colored window shows ground truth anomalous region. (a), (b), (c) and (d) show videos containing *animal abuse (beating a dog)*, *explosion*, *road accident* and *shooting*, respectively. (e) and (f) show normal videos with no anomaly. (g) and (h) present two failure cases of our anomaly detection method.

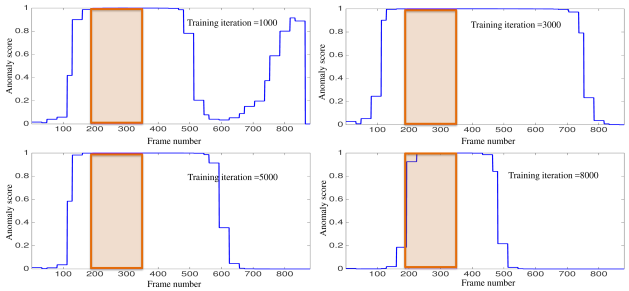

Figure 8. Evolution of score on a training video over iterations. Colored window represents ground truth (anomalous region). As iteration increases, our method generates high anomaly scores on anomalous video segments and low scores on normal segments.

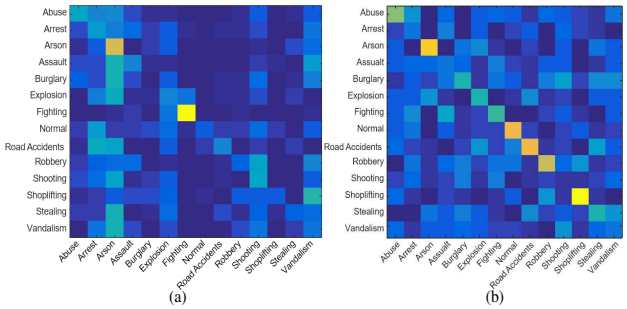

Figure 9. (a) and (b) show the confusion matrices of activity recognition using C3D [37] and TCNN [21] on our dataset.

Method	C3D [37]	TCNN [21]
Accuracy	23.0	28.4

Table 5. Activity recognition results of C3D [37] and TCNN [21].

lous videos during data collection, but which are not used for our anomaly detection method discussed above. For activity recognition, we use 50 videos from each event and divide them into 75/25 ratio for training and testing. We provide two baseline results for activity recognition on our dataset based on a 4-fold cross validation. For the first baseline, we construct a 4096-D feature vector by averaging C3D [37] features from each 16-frames clip followed by an L2-normalization. The feature vector is used as an input to a nearest neighbor classifier. The second baseline is the Tube Convolutional Neural Network (TCNN) [21], which introduces the tube of interest (ToI) pooling layer to replace the 5-th and 3d-max-pooling layer in C3D pipeline. The ToI pooling layer aggregates features from all clips and outputs one feature vector for a whole video. The quantitative results *i.e.* confusion matrices and accuracy are given in Figure 9 and Table 5. These state-of-the-art action recognition methods perform poor on this dataset. It is because the videos are long untrimmed surveillance videos with very large intra-class variations. Therefore, our dataset is a unique and challenging dataset for anomalous activity recognition.

6. Conclusions

We propose a deep learning approach to detect real-world anomalies in surveillance videos. Due to the complexity of these realistic anomalies, using only normal data alone may not be optimal for anomaly detection. We attempt to exploit both normal and anomalous videos. To avoid labor-intensive temporal annotations of anomalous segments in training videos, we learn a general model of anomaly detection using deep MIL framework with weakly labeled data. To validate the proposed approach, a new large-scale anomaly dataset consisting of a variety of real-world anomalies is introduced. The experimental results on this dataset show that our proposed anomaly detection approach performs significantly better than baseline methods. Furthermore, we demonstrate the usefulness of our dataset for the task of anomalous activity recognition.

Acknowledgement. The project was supported by Award No. 2015-R2-CXK025, awarded by the National Institute of Justice, Office of Justice Programs, U.S. Department of Justice. The opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect those of the Department of Justice.

References

[1] http://www.multitel.be/image/research-development/research-projects/boss.php.

[2] Unusual crowd activity dataset of university of minnesota. In *http://mha.cs.umn.edu/movies/crowdactivity-all.avi.*

[3] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *TPAMI*, 2008.

[4] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, pages 577–584, Cambridge, MA, USA, 2002. MIT Press.

[5] B. Anti and B. Ommer. Video parsing for abnormality detection. In *ICCV*, 2011.

[6] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016.

[7] A. Basharat, A. Gritai, and M. Shah. Learning object motion patterns for anomaly detection and improved object detection. In *CVPR*, 2008.

[8] C. Bergeron, J. Zaretzki, C. Breneman, and K. P. Bennett. Multiple instance ranking. In *ICML*, 2008.

[9] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 2009.

[10] X. Cui, Q. Liu, M. Gao, and D. N. Metaxas. Abnormal detection using interaction energy potentials. In *CVPR*, 2011.

[11] A. Datta, M. Shah, and N. Da Vitoria Lobo. Person-on-person violence detection in video data. In *ICPR*, 2002.

[12] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997.

[13] S. Ding, L. Lin, G. Wang, and H. Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 48(10):2993–3003, 2015.

[14] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 2011.

[15] Y. Gao, H. Liu, X. Sun, C. Wang, and Y. Liu. Violence detection using oriented violent flows. *Image and Vision Computing*, 2016.

[16] A. Gordo, J. Almazán, J. Revaud, and D. Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016.

[17] M. Gygli, Y. Song, and L. Cao. Video2gif: Automatic generation of animated gifs from video. In *CVPR*, June 2016.

[18] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis. Learning temporal regularity in video sequences. In *CVPR*, June 2016.

[19] G. E. Hinton. Rectified linear units improve restricted boltzmann machines vinod nair. In *ICML*, 2010.

[20] T. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behaviour in video. In *ICCV*, 2009.

[21] R. Hou, C. Chen, and M. Shah. Tube convolutional neural network (t-cnn) for action detection in videos. In *ICCV*, 2017.

[22] T. Joachims. Optimizing search engines using clickthrough data. In *ACM SIGKDD*, 2002.

[23] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE Transactions on Intelligent Transportation Systems*, 1(2):108–118, 2000.

[24] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[25] J. Kooij, M. Liem, J. Krijnders, T. Andringa, and D. Gavrila. Multi-modal human aggression detection. *Computer Vision and Image Understanding*, 2016.

[26] L. Kratz and K. Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *CVPR*, 2009.

[27] W. Li, V. Mahadevan, and N. Vasconcelos. Anomaly detection and localization in crowded scenes. *TPAMI*, 2014.

[28] C. Lu, J. Shi, and J. Jia. Abnormal event detection at 150 fps in matlab. In *ICCV*, 2013.

[29] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *CVPR*, 2009.

[30] S. Mohammadi, A. Perina, H. Kiani, and M. Vittorio. Angry crowds: Detecting violent events in videos. In *ECCV*, 2016.

[31] H. Rabiee, J. Haddadnia, H. Mousavi, M. Kalantarzadeh, M. Nabi, and V. Murino. Novel dataset for fine-grained abnormal behavior understanding in crowd. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2016.

[32] I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *TPAMI*, 31(8):1472–1485, 2009.

[33] A. Sankaranarayanan, S. Alavi and R. Chellappa. Triplet similarity embedding for face verification. *arXiv preprint arXiv:1602.03418*, 2016.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 2014.

[35] W. Sultani and J. Y. Choi. Abnormal traffic detection using intelligent driver model. In *ICPR*, 2010.

[36] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688*, 2016.

[37] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

[38] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014.

[39] S. Wu, B. E. Moore, and M. Shah. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In *CVPR*, 2010.

[40] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe. Learning deep representations of appearance and motion for anomalous event detection. In *BMVC*, 2015.

[41] T. Yao, T. Mei, and Y. Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *CVPR*, June 2016.

[42] B. Zhao, L. Fei-Fei, and E. P. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *CVPR*, 2011.

[43] Y. Zhu, I. M. Nayak, and A. K. Roy-Chowdhury. Context-aware activity recognition and anomaly detection in video. In *IEEE Journal of Selected Topics in Signal Processing*, 2013.