Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 115 (2017) 448–459

Procedia Computer Science

www.elsevier.com/locate/procedia

7th International Conference on Advances in Computing & Communications, ICACC-2017, 22-24 August 2017, Cochin, India

Video Anomaly Detection in Confined Areas

Emmanu Varghese, Jaison Mulerikkal*, Amitha Mathew

Rajagiri School of Engineering and Technology, Rajagiri Valley, Kochi, 682039, India

Abstract

This paper proposes a new supervised algorithm for detecting abnormal events in confined areas like ATM room, server room etc. In the training phase, algorithm learns the motion path and speed of objects in the video. In the testing phase, if any motion happens other than in the learned motion path or the speed of object has large variation from the learned speed then the algorithm alert it as abnormal event. The proposed method process video in groups of frames. The algorithm uses statistical functions to learn the motion path and speed of objects in a video.

Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications.

Keywords: Video anomaly detection; Dense sampling; Colour pattern; Gray scale value; Statistical methods

1. Introduction

Nowadays, the popularity of surveillance cameras is increasing. There are security cameras in public places like railway station, airport etc. Private organizations also has security cameras in their premises to deal with security challenges like robbery or terrorist attack. From these security cameras, detecting and alerting abnormal events manually requires a security person solely employed for this purpose. The accuracy of manual monitoring depends on the skill and concentration of that security person. In confined areas like ATM room, server room etc., there are chances of mysterious events like robbery, destroying the machine etc. This paper proposes a new method to detect and alert such abnormal events happening in confined areas without human intervention.

There are many research papers about detecting anomalies in a video. Previous works happened in this area can be generally classified in to two categories, supervised and unsupervised methods. Supervised methods have a training phase [1] and unsupervised methods don't have such an explicit training phase. This paper proposes a new supervised video anomaly detection algorithm with less mathematical computation than existing algorithms. The term abnormal event in this paper is meant by the events that have no similarity with the events that are used for training. If a video that contains a person is walking is used for training then the abnormal events can be the person not walking or runs

* Corresponding author. Tel.: +91 484 2660999 ; fax: +91 484 2426241.
E-mail address: jaisonmpaul@rajagiritech.edu.in

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications
10.1016/j.procs.2017.09.104

very fast. In this paper, objects in a video is meant by the things or the persons included in the video. If the video is from the camera of an ATM room then the objects are the person who uses the ATM machine and the ATM machine itself. If the video is from the camera near to a gate in car parking area then the objects are the car that passes through the gate and the gate itself. Video event detection is the process of detecting events in a video. Video anomaly detection is the process of detecting abnormal events in a video. Video anomaly detection is a subclass of video event detection.

The aim of this paper is to introduce a new supervised video anomaly detection algorithm for confined areas using the colour pattern of the videos. Our work learns the motion path and speed of objects in the training video by processing the video in ensembles [2]. By analysing the motion path and amount of motion in training video, abnormal events can be detected from the testing videos. At training phase algorithm learns the allowed areas in the video where motion is permitted and maximum and minimum speed of that motion.

2. Related work

Detecting abnormal events in video is an active research area and so many works had done in this area. As discussed earlier, video event detection can be broadly classified into supervised and unsupervised. The method described in this paper follows supervised video event detection.

For video event detection, video is processed in group of frames because an event always spans many frames so processing video frame by frame is meaningless. An unsupervised method of abnormal event detection is described in [2]. A method of dense sampling and spatio-temporal volume construction is described in the work of [2] in which group of frames is taken and is divided into spatio-temporal volumes and collection of such volumes called ensemble is taken for processing. Then the probability of each ensemble to be normal is computed. If an ensemble of the video has large variation in probability than the computed one then that ensemble is marked as abnormal ensemble. Another unsupervised deep learning framework for irregular occasion recognition in complex video scenes is depicted in [9]. They propose Appearance and Motion DeepNet (AMDN) which utilizes deep neural networks to consequently learn feature representations. Anomaly score of each representations is predicted using various one-class Support Vector Machine(SVM) models. This anomaly score is further used for anomaly detection. A framework for detecting abnormal events in crowded scenes is depicted in [10], where common behaviour of a scene is identified by modelling the variations of local spatio-temporal motion patterns. Then the spatial and temporal relationships between this motion patterns is found to characterize the behaviour of the entire sequence and the unusual events are identified as statistical deviations in video sequences of the same scene. Convolution networks can be used to extract good features from video. In [11], they show that 3D convolutional deep networks are good feature learning machines that model appearance and motion simultaneously.

Boiman and Irani [3] describe the problem of detecting anomalies as the problem of creating the new image using spatio-temporal patches extracted from previous images. The patches of training images are stored in the database during training and in the testing phase, if the testing image can be constructed using large contiguous chunks of data from the database, it is considered as normal. The testing image that cannot be created from the database is regarded as abnormal [3]. Figure 1 illustrates the method proposed by Boiman and Irani . A method to detect abnormal events happening inside ATM room is described in the work of [4] in which motion history image (MHI) and Hu moments are used to extract relevant features from video. The dimensionality of the features are reduced by Principle Component Analysis (PCA). Normal and abnormal events are classified using SVM. A supervised video anomaly detection method is described in [13] where motion descriptors are first extracted and quantized into small blocks. Spatio-temporal filters at various scales are applied to found the smooth estimates at each spatio-temporal location for each feature descriptor. Local K-Nearest Neighbors(KNN) distance for each location is computed for training and testing video. These neighbourhood KNN distances are totaled to create a composite score for the test and training video. The melded scores are ranked to determine anomalies.

An anomaly detection system using low-level features are described in [5] where dense motion field and statistics are computed in each frame. Then motion directional PCA technique is used to extract useful principle features in time-span. Finally one-class SVM discriminates the anomaly from normal events. A method to detect abnormal events at three levels considering spatio-temporal context of video objects is proposed in [12]. The three levels namely point anomaly, sequential anomaly and co-occurrence anomaly. Point anomaly is the anomaly caused by the instant behaviour of a single object in the video. Sequential anomaly is the anomaly caused by two or more instantaneous

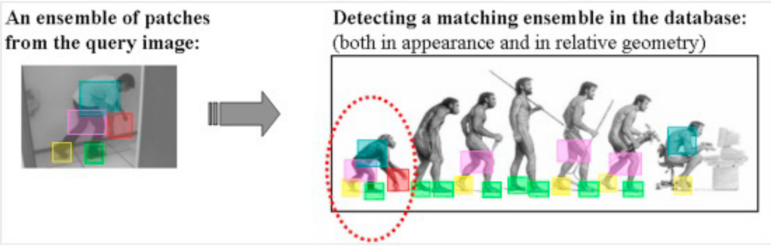


Fig. 1: Detecting a matching ensemble of patches [3].

behaviour of an object. Co-occurrence anomaly is the anomaly caused by the instantaneous behaviour of multiple objects.

The method proposed in this paper uses a different strategy to identify abnormal events in confined areas. In this method, the colour pattern of the training video and testing videos are compared to identify abnormal events. From the literature survey, we found that in supervised video anomaly detection methods, abnormal events are events that are not similar to normal events that are used for training. We noticed that when an abnormal event happened in a video, a large variation in colour pattern is occurring in some part of the video relative to the colour pattern in that part of the training video. The variation in colour pattern is more significant in videos that are shot in confined areas. The method proposed in this paper is emanated by the research done on how this variation in colour pattern happening in different parts of the video can be used to identify abnormal events? The proposed method uses a modified version of dense sampling technique mentioned in [2] to process video in ensembles. There are two phases in this method, training phase and testing phase. In training phase, the proposed method divide video in to ensembles and for each ensemble gray scale colour pattern is found. From this colour pattern, the motion path and speed of objects is learned using statistical methods. In the testing phase, the video is divided in to ensembles and the motion path and speed of objects in the video is found as in the training phase. If the motion path and speed of objects in the testing video has large variations from that of training video, notification for abnormal event is triggered.

3. The proposed method

The proposed method has two phases, training phase and testing phase. In the training phase, algorithm learns the motion path and speed of objects in the training video. In the testing phase, if any motion happens other than in the learned motion path or the speed of objects has large variation from that of learned speed then the algorithm alert it has anomaly. Motion path is learned using the pixel's gray scale value and statistical functions minimum and maximum. Pixel's gray scale value and statistical function mode are used to get an idea about object's speed. In order to identify if there is motion in the video after the allowed time for an event, algorithm uses a simple motion detection technique that is described in section 3.2.1. When training is completed the algorithm understands the motion path and speed of objects in the training video. In the testing phase

1. If any motion happens in the learned path and speed of object is similar to the learned speed then it is a normal event.
2. If any motion happens in the learned path and speed of object has large difference from the learned speed then it is an abnormal event.
3. If any motion happens in an area other than the learned path then that it is detected as abnormal event.
4. If there is motion after the allowed time for that event then it is detected as abnormal event.

In order to process the video, the video is first converted to gray scale then a modified version of dense sampling technique in [2] is used where a group of frames is taken and is sampled in to many spatio-temporal volumes as shown in the figure 2. In [2], around each pixel a small volume of size $nx \times ny \times nt$ is created called spatio-temporal volumes in which $nx \times ny$ is the size of the spatial (image) window and nt is the depth of the video volume in time, then around each pixel a much larger region of size $px \times py \times pt$ is considered, in which $px > nx$, $py > ny$ and $pt > nt$. So in [2],

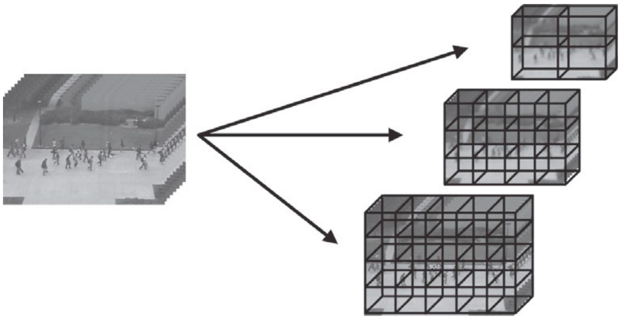


Fig. 2: Dense sampling[2]

an ensemble consist of pixels and around each pixel there is spatio-temporal volume. This whole work [2] can be considered as an extension of the Bag of Video words (BOV) approach. The method described in [2] imposes spatial and temporal constraints on the video volumes so that an inference mechanism can estimate the probability density functions of their arrangements. Anomalous events are assumed to be video arrangements with very low frequency of occurrence. A modified version of the dense sampling described in [2] is used in the proposed algorithm. The method of dense sampling used in this paper consist of the following steps.

1. Group a number of frames into a pack for e.g. take a group of 25 frames as a pack.
2. Starting from top left, divide that pack in to larger blocks called ensembles as shown in the figure 3.
3. That ensemble is again divided in to smaller blocks called spatio-temporal volumes.

For example, if number of frames inside a pack is 5 and frame size is 640×480, we can divide it into ensembles of size 80×80(80 pixel width and height) and that ensembles can be divided into spatio-temporal volumes of size 8×8. So there will be 48 ensembles and inside each ensemble there will be 100 spatio-temporal volumes as shown in Fig 3. The algorithm for training is shown in algorithm 1 and that for testing is shown in algorithm 2. The computational complexity to check whether a spatio-temporal volume is normal or not is $O(MN)$. Where M is the number of pixels inside a spatio-temporal volume in a single frame and N is the the number of frames in a pack. In the example mentioned above M=64 and N=5.

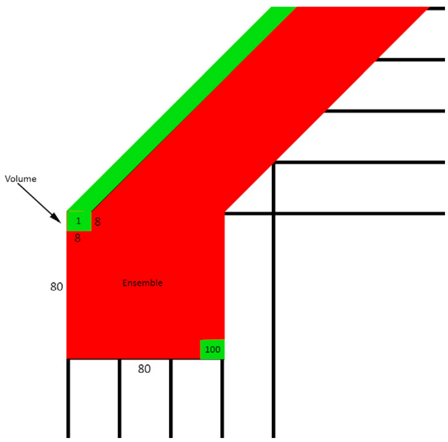


Fig. 3: Dense sampling if frame size is 640×480, ensembles size is 80×80 and spatio-temporal volume size is 8×8

3.1. Training

For training, we require the video sample of an event to be normal. If the proposed system is intended to work in an ATM room then the normal video sample consist of a person taking cash from an ATM machine without causing damage to the ATM machine. If the proposed system is intended to work in a gate then the normal video sample consist of a person passing through by opening and closing the gate in a normal speed without jumping or destroying the gate. The algorithm for training is shown in algorithm 1.

Algorithm 1 Algorithm for training

1: Start
2: Read training video and capture frames.
3: Divide the captured video frames to packs of N frames, find the total number of packs (TN) in the training video, number of frames in a pack depends on the computation power of the computer where the algorithm is intended to work.
4: For each pack

   1. Perform dense sampling and divide the pack into ensembles and ensemble into sptio-temporal volumes.
   2. Find the Frame Volume Gray Scale Value(FVGV) of all volumes. FVGV of a volume is found by summing the gray scale values of all pixel inside that volume in a single frame.
   3. Find the Single Volume Gray Scale Max Value(SMaV) and Single Volume Gray Scale Min Value(SMiV) of all volumes. SMaV and SMiV of a volume is found by taking the maximum and minimum of the FVGV values for that volume across the pack.

5: Find the Gray Scale Max Threshold(GMaT) and Gray Scale Min Threshold(GMiT) of all volumes. GMaT and GMiT of a volume is found by taking the maximum and minimum of the SMaV and SMiV values respectively for that volume across the whole number of packs. Store the GMaT and GMiT values of all volumes.
6: Find mode gray scale values(Mode_max and Mode_min) of all volumes. Mode gray scale values of a volume is found by taking the mode of the SMaV and SMiV for that volume across the whole number of packs. Store these values as the Threshold Mode Max(TMMa) and Threshold Mode Min(TMMi) values for that volume.
7: Stop

The mathematical equations for finding the FVGV, SMaV, SMiV and mode gray scale values are as follows

$$\text{FVGV(Frame Volume Gray Scale Value)} = \sum_{i=1}^{S} G_i \tag{1}$$

Where G_i is the Gray scale value of a pixel at location i inside a volume in a single frame and S is the total number of pixels in a volume

$$\begin{aligned} \text{SMaV(Single Volume Gray Scale Max Value)} &= \max_{1\le i\le N}\{FVGV_i\} \\ \text{SMiV(Single Volume Gray Scale Min Value)} &= \min_{1\le i\le N}\{FVGV_i\} \end{aligned} \tag{2}$$

Where N is the total number of frames in a pack and $FVGV_i$ is the FVGV of the volume in ith frame.

$$\begin{aligned} \text{GMaT(Gray Scale Max Threshold)} &= \max_{1\leq i\leq TN}\{SMaV_i\} \\ \text{GMiT(Gray Scale Min Threshold)} &= \min_{1\leq i\leq TN}\{SMiV_i\} \end{aligned} \tag{3}$$

$$\begin{aligned} \text{Mode\_max} &= \underset{1\leq i\leq TN}{\text{mode}}\{SMaV_i\} \\ \text{Mode\_min} &= \underset{1\leq i\leq TN}{\text{mode}}\{SMiV_i\} \end{aligned} \tag{4}$$

In Equation 3 and 4, TN is the total number of packs in the training video and $SMaV_i$ and $SMiV_i$ is the SMaV and SMiV of the volume in ith pack.

Consider the example scenario where the frame size is 640×480, ensembles size is 80×80(80 pixel width and 80 pixel height), spatio-temporal volumes of size 8×8, the number of frames grouped is 5 and total number of packs is 100. Inside a spatio-temporal volume, in a frame there will be 64 pixels and in the whole spatio-temporal volume there will be 320 pixels(8×8 in a single frame, frames are grouped to 5 so total $8\times 8\times 5$ pixels). Then the sum of gray scale value of all pixels inside spatio-temporal volume in a single frame is computed using the equation 1 as FVGV value of that spatio-temporal volume in that frame. Then the maximum and minimum of such values inside that volume is computed using the equation 2 and will assign that values as the SMaV and SMiV value of that volume. In this case, for each spatio-temporal volume there will be 5 FVGV values and we take the maximum and minimum of that 5 values. Considering total number of packs(100), the maximum of SMaV and minimum of SMiV is computed for each volume using the equation 3 and assign that values as GMaT and GMiT of that volume respectively. The mode values of SMaV and SMiV for all volumes across the total number of packs(100) is computed using the equation 4 and store these values as threshold mod values of those volumes.

3.2. Testing

The algorithm for testing is described in algorithm 2. In dense sampling video is divided into ensembles and in the testing mode we are finding out whether an ensemble is normal or abnormal. Depending on the resolution of the video, number of ensembles will change. An event is abnormal, if the number of abnormal ensembles is more than ensemble_threshold. If the value of ensemble_threshold is very low, the algorithm will be very sensitive to abnormal events and a minor change from the training video will be detected as abnormal event. If the value of ensemble_threshold is very high, the algorithm will be less sensitive to abnormal events and only events that are totally different from training video will be detected as abnormal events. According to the situation where the algorithm intended to work, we can change the sensitivity of the algorithm by adjusting the value of ensemble_threshold. The other sensitivity controlling parameters are volume_threshold and mode_threshold which are described in the algorithm 2. The sensitivity of the algorithm is inversely proportional to ensemble_threshold, volume_threshold and mode_threshold.

3.2.1. Motion detection

Whenever the number of packs captured become greater than TN, where TN is the total number of packs in the training video, a simple motion detection technique is executed to check whether there is motion or not after TN packs. If there is continuous motion after TN packs, it implies that the event is taking more time than the allowed time for that event and it is labelled as abnormal event. After TN packs, if there is no motion for RT packs, where RT is the Reinitialization Time, it implies that the event has finished and algorithm 2 is reinitialized. The variable RT depends on the situation where the algorithm intended to work. A motion is happened in the video when there is a change in the position of an object relative to its surroundings or a change in the surroundings relative to an object. For motion detection we are using frame difference method described in [6], where the intensity of a pixel in two consecutive frames are compared to identify if any motion had happened to that pixel.

Algorithm 2 Algorithm for testing

Start

Read testing video and start capturing frames.

When number of frames captured is N (N is the number of frames in a pack)

1. Perform dense sampling and divide the pack into ensembles and ensemble into spatio-temporal volumes.
2. Find FVGV, SMaV, SMiV of all volumes.
3. A volume is abnormal if its SMaV value greater than the GMaT value for that volume or SMiV value less than GMiT value for that volume.
4. If the number of volumes in an ensemble that are abnormal is greater than volume_threshold. Then that ensemble is abnormal.

When number of packs captured is TN, where TN is the total number of packs in the training video.

1. Find mode gray scale values(Mode_max and Mode_min) of all volumes.
2. A volume is abnormal if its Mode_max is greater than TMMa for that volume or Mode_min is less than TMMi for that volume.
3. If the number of volumes in an ensemble that are abnormal is greater than mode_threshold then that ensemble is abnormal.

If violated ensemble count is more than ensemble_threshold, then the processed pack contain abnormal event and notification for abnormal event is triggered.

Stop

4. Experiments

The proposed algorithm is implemented using python 2.7 and OpenCV 2.4 on a windows 10 computer having Intel core i3, 1.9 GHz processor with 4 GB RAM. The algorithm is trained and tested using videos shot by ourself[1] with resolution 640×480. The sensitivity parameters ensemble_threshold, volume_threshold and mode_threshold are set to 5, 50 and 50 respectively because at these values algorithm works with reasonable sensitivity. At these values algorithm detected abnormal events that had perceptible difference from normal events. The rational behind these values is explained in section 4.3.

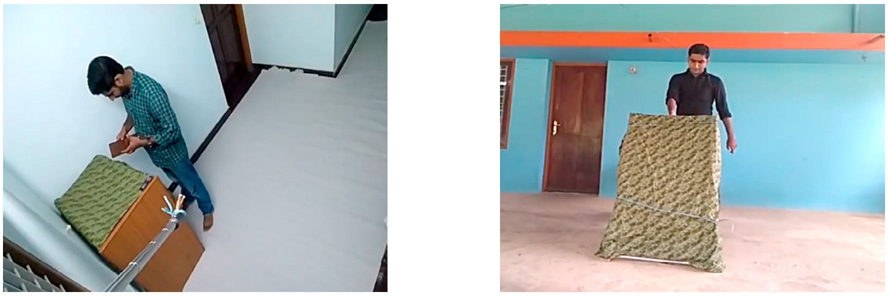
4.1. Dataset

The proposed algorithm identifies abnormal events by comparing the colour pattern of testing videos with training videos. The training and testing videos must be shot in same environment and using the camera that is placed at same place and angle because a minor change in any of these will affect colour pattern of the video. We created our own dataset for testing the algorithm. The many standard video datasets available misses the specific criteria of the proposed algorithm that is the training and testing video must be shot in same conditions as mentioned above. We trained and tested the algorithm in two datasets that are created by ourself. The first dataset contain 5 videos of different events happening inside an ATM room and the second dataset contain 4 videos of different events happening at the gate. Inside each dataset there is a training video and a set of testing videos. We created ATM room and entry gate like environments and videos containing normal and abnormal events are shot by ourself. For each dataset, the training video and testing videos was shot by the camera that is placed at the same place and angle because a minor change in place or angle will affect colour pattern of the video. The algorithm shown good results in detecting abnormal events.

[1] Video will be available on request for research and verification purpose at https://goo.gl/pWfGXb

4.2. Experimental results and comparison

The proposed method detected abnormal events in the testing videos with a false positive. The performance of the proposed method is evaluated and compared with other methods [4], [5] and [13] in terms of computational complexity. The screenshots of taining videos at ATM room and entry gate are shown in Fig 4a and Fig 4b respectively. The algorithm detected abnormal events from the testing videos. If two people enters the ATM room, the algorithm detected it as abnormal event. If the two people entered are adults, it is an abnormal event. But there are normal situations in real life where two people enters the ATM room. If a mother and child enters the ATM room, the system will detect it as abnormal event but it is a false positive. The screenshots of the testing videos when processed through the algorithm are shown in Fig 5 and Fig 6.



(a) ATM room (b) Entry gate

Fig. 4: Training videos for ATM room and entry gate

(a) Two persons entering ATM room (b) Robbery inside ATM room (c) Damaging ATM machine

Fig. 5: ATM room testing videos when processed through the algorithm

(a) Jumping over the gate (b) Damaging the gate

Fig. 6: Entry gate testing videos when processed through the algorithm

4.2.1. *ATM room*

In this dataset, the training video consist of a single person entering the ATM room, using the ATM machine and leaving the room after usage without taking too much time or damaging the machine. The events in the testing videos and the algorithm output are depicted in table 1.

Table 1: ATM room testing dataset

Event	Result
Single person entering the ATM room, using ATM machine and leaving the room after usage without taking too much time and damaging the machine	Detected as normal event
Two people entering the ATM room, using ATM machine and leaving the room after usage without taking too much time and damaging the machine	Detected as abnormal event
Robbery inside the ATM Room. A single person taking cash from ATM machine. The thief enters the ATM room, beats the person and steal money from him	Detected as abnormal event
Single person entering the ATM room and damaging the machine	Detected as abnormal event
Single person entering the ATM room and spending too much time inside ATM room	Detected as abnormal event

4.2.2. *Entry gate*

In this dataset, the training video consist of a single person placing his credentials at gate and passing through the gate. The events in the testing videos and the algorithm output are depicted in table 2.

Table 2: Entry gate testing dataset

Event	Result
Single person placing his credentials at gate and passing through the gate without taking too much time and damaging the gate	Detected as normal event
Single person jumping over the gate without placing his credentials	Detected as abnormal event
A single person passes through the gate by damaging the gate	Detected as abnormal event
A single person takes too much time to pass through the gate	Detected as abnormal event

4.2.3. *Performance analysis*

Performance evaluation of this anomaly detection algorithm is conducted at the developing computer. The time taken for training videos of different time period is shows in table 3.

In the testing phase, algorithm processed 3 packs in a second. Since the result is coming after processing each pack, the algorithm detected abnormal events with a delay of 0.333 seconds or 333 milliseconds. The computational complexity of this algorithm to check whether a spatio-temporal volume is abnormal or not is $O(MN)$. Where M is the number of pixels inside a spatio-temporal volume in a single frame and N is the the number of frames in a pack. There is no high level feature extraction and data classification algorithms involved in this method. The proposed method takes the pixel's gray scale value as low-level feature and no dimensionality reduction techniques like PCA is used.

Table 3: Training time for videos of different length

Training video time	Time taken to train
2 minutes	4.05 minutes
1 minutes	2 minutes
30 seconds	58 seconds

This is a great advantage of this method over [4], [5] and [13]. The method proposed in [4] uses motion history image and Hu moments to extract relevant features from video then they use PCA for dimensionality reduction and SVM for data classification. The method proposed in [5] uses motion vector statistics as features and then they use PCA for dimensionality reduction and SVM for data classification. Computational complexity of PCA [7] with n data points and each represented with p features is $O(min(p^3, n^3))$ and that of SVM(LibSVM) [8] to classify n numbers of data is $O(n^3)$. The method proposed in [13] uses KNN and the computational complexity of KNN with n data points and each represented with d features, k closest points is $O(nk+nd)$. Since our method does not contain PCA computation, SVM or KNN, we reasonably assume that our method has less computational complexity than the other methods discussed here.

The accuracy of this method is determined by analysing how many alerts for abnormal event is triggered when considering total number of abnormal packs in an event. The algorithm process video in packs and after processing each pack, the algorithm will output whether the processed pack is abnormal or not. The accuracy is computed using the equation 5. When computing accuracy, we are not considered false positives. The events robbery inside the ATM room and damaging the gate are chosen for accuracy calculation. Table 4 and 5 shows the accuracy of the algorithm at different ensemble_threshold in ATM room and entry gate environments respectively. Since we not used standard datasets, we are not able the compare the accuracy of our algorithm with other existing methods. However in the dataset we have created, algorithm shown good accuracy.

$$\text{Accuracy} = \frac{Number\ of\ detected\ abnormal\ packs}{Total\ number\ of\ abnormal\ packs} \tag{5}$$

Table 4: Accuracy of the algorithm in ATM room

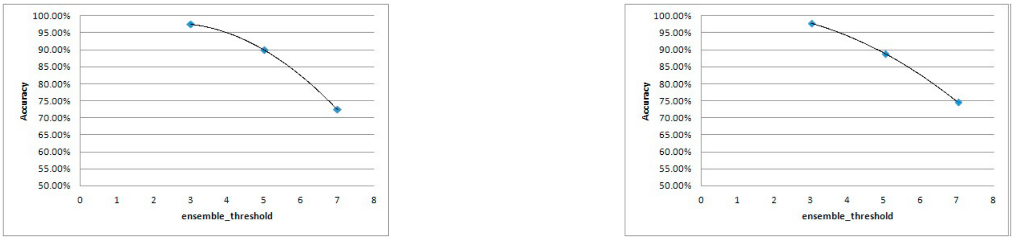
ensemble_threshold	3	5	7
Total number of abnormal packs	40	40	40
Detected number of abnormal packs	39	36	29
Accuracy	97.5%	90%	72.5%

Table 5: Accuracy of the algorithm in entry gate

ensemble_threshold	3	5	7
Total number of abnormal packs	91	91	91
Detected number of abnormal packs	89	81	68
Accuracy	97.80%	89.01%	74.72%

4.3. Sensitivity parameters

There are three sensitivity parameters ensemble_threshold, volume_threshold and mode_threshold. The accuracy of the algorithm at various ensemble_threshold is shown graphically in Fig 7. We can see from the graph that the sensitivity of the algorithm is inversely proportional to ensemble_threshold. Similar to ensemble_threshold, the sensitivity of the algorithm is inversely proportional to volume_threshold and mode_threshold.



(a) ATM room (b) Entry gate

Fig. 7: Sensitivity of the algorithm at various ensemble_threshold values

The optimal value of ensemble_threshold in the experiments we have conducted is 5 because if the value of ensemble_threshold is less than 5, the algorithm will be highly sensitive. If the value of ensemble_threshold is greater than 5, the algorithm will be less sensitive. When the value of ensemble_threshold is 5, algorithm detected abnormal events that had perceptible difference from normal events. Similarly, if the value of volume_threshold and mode_threshold is less than 50, the algorithm will be highly sensitive. if the value of volume_threshold and mode_threshold is greater than 50, the algorithm will be less sensitive. When the value of volume_threshold and mode_threshold is 50, algorithm detected abnormal events that had perceptible difference from normal events.

5. Conclusion and future scope

In this paper, we have proposed a new algorithm for video anomaly detection in confined areas. Dense sampling is used to convert the video to spatio-temporal volumes and ensembles. Algorithm uses this spatio-temporal volumes and ensembles to process the video. The proposed algorithm uses pixel's gray scale value as low-level feature. Statistical functions maximum, minimum and mode are employed for training and testing. Our algorithm accurately detected abnormal events from testing videos. The proposed algorithm has less computational complexity than existing algorithms. There are sensitivity controlling parameters that can be adjusted according to the situation where the algorithm intended to work. As a future scope, a good image classification tool like Google TensorFlow can be used to identify what abnormal event had occurred.

References

[1] Peyrard, Nathalie, and Patrick Bouthemy. (2003) "Detection of meaningful events in videos based on a supervised classification approach." *International Conference on Image Processing* 3:621-624.
[2] Roshtkhari, Mehrsan Javan, and Martin D. Levine. (2013) "An on-line, real-time learning method for detecting anomalies in videos using spatio-temporal compositions." *Computer vision and image understanding* 117(10):1436-1452.
[3] Boiman, Oren, and Michal Irani. (2005) "Detecting irregularities in images and in video." *IEEE International Conference on Computer Vision* 1:462-469.
[4] Tripathi, Vikas, Durgaprasad Gangodkar, Vivek Latta, and Ankush Mittal. (2015) "Robust Abnormal Event Recognition via Motion and Shape Analysis at ATM Installations." *Journal of Electrical and Computer Engineering* 2015:1-10.
[5] Liu, Chang, Guijin Wang, Wenxin Ning, Xinggang Lin, Liang Li, and Zhou Liu. (2010) "Anomaly detection in surveillance video using motion direction statistics." *IEEE International Conference on Image Processing* 717-720.
[6] Rosebrock, A. (2017). *Basic motion detection and tracking with Python and OpenCV - PyImageSearch.* [online] PyImageSearch. Available at: http://www.pyimagesearch.com/2015/05/25/basic-motion-detection-and-tracking-with-python-and-opencv/ [Accessed 17 Apr. 2017].
[7] Johnstone, Iain M., and Arthur Yu Lu. (2009) "On consistency and sparsity for principal components analysis in high dimensions." *Journal of the American Statistical Association* 104(486):682-693.

[8] Abdiansah, Abdiansah, and Retantyo Wardoyo. (2015) "Time complexity analysis of support vector machines (SVM) in LibSVM." *International Journal Computer and Application* 128(3):28-34.

[9] Xu, Dan, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. (2015) "Learning deep representations of appearance and motion for anomalous event detection." *arXiv preprint arXiv:1510.01553*.

[10] Kratz, Louis, and Ko Nishino. (2009) "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models." *IEEE Conference on Computer Vision and Pattern Recognition* 1446-1453.

[11] Tran, Du, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. (2015) "Learning spatiotemporal features with 3d convolutional networks." *IEEE International Conference on Computer Vision* 4489-4497.

[12] Jiang, Fan, Junsong Yuan, Sotirios A. Tsaftaris, and Aggelos K. Katsaggelos. (2010) "Video anomaly detection in spatiotemporal context." *IEEE International Conference on Image Processing* 705-708.

[13] Saligrama, Venkatesh, and Zhu Chen. (2012) "Video anomaly detection based on local statistical aggregates." *IEEE Conference on Computer Vision and Pattern Recognition* 2112-2119.