Unusual Activity Analysis using Video Epitomes and pLSA

Ayesha Choudhary[1] Manish Pal[1] Subhashis Banerjee[1] Santanu Chaudhury[2]
[1]*Dept. of Computer Science and Engineering,* [2]*Dept. of Electrical Engineering*
Indian Institute of Technology, Delhi
[1]*ayesha, suban@cse.iitd.ernet.in,* [1]*manishpal07@gmail.com,* [2]*santanuc@ee.iitd.ernet.in*

Abstract

In this paper, we address the problem of unsupervised learning of usual patterns of activities in an area under surveillance and detecting deviant patterns. We use video epitomes for segmenting foreground objects from background and obtain an approximate shape, trajectory and temporal information in the form of space-time patches. We apply pLSA for finding correlations among these patches to learn usual activities in the scene. We also extend pLSA to classify a novel video as usual or unusual.

1. Introduction

In this paper, we address the problem of unsupervised learning of usual patterns of activities and detecting deviant patterns from a surveillance video. The activity patterns are described by the shape and trajectories of moving objects in the video sequences. We use video epitomes [3] to segment the foreground object in the form of space-time patches and pLSA (probabilistic Latent Semantic Indexing) [5] to find spatio-temporal correlations among these patches. The spatial correlations among the patches in a frame define the approximate shape of the object while the spatio-temporal correlations across frames describes the trajectory. We also use pLSA to discover landmarks and correlations among these in the area under surveillance. By landmarks, we mean common locations at which people enter, exit or are static. The correlations among these landmarks provide information about the common paths taken in the scene. The main contributions of our work are:

1. Segmentation of foreground objects from low resolution video epitomes, that gives us both the approximate shape and the trajectory of the objects in the form of space-time patches (4.2).

2. Spatio-temporal correlations among related patches using pLSA for learning usual activity patterns in the video without performing high reolution segmentation(4.3).

3. Correlations among short tracks using pLSA to find the common paths and landmarks in the scene (4.3).

4. Extension of pLSA for classifying a novel video as usual or unusual.

In our framework, we use video epitomes [3] because an epitome represents the complete original video at a low resolution and captures nearly all the spatial and temporal information of the video. Moreover, a video epitome is the smallest compressed form of a video from which the complete video can be reconstructed and is also capable of reconstructing lost frames. It is to be noted that by using video epitomes for segmentation, the complete shape, trajectory and temporal information of the foreground objects in the scene can be obtained in the form of space time patches, inspite of lost frames in the original video.

We use pLSA for finding spatio-temporal correlations among the space time foreground patches and also among the landmarks and use these correlations to model the usual activities in the scene. The spatio-temporal correlations give us the approximate shape and trajectory of the foreground object. Moreover, pLSA also correlates short broken tracks on the same path and overcomes the shortcomings of tracking based unusual activity detection methods, that fail when the tracks are not complete. This also gives us the ability to break the video into fixed sized clips randomly, where the clips need not contain the complete trajectory of an object. This makes the system robust with respect to small broken tracks which occur due to undetected features and variations in segmented shape of foreground object. It also enables detecting unusual activities based on both the shape and the trajectory of the object without performing high resolution segmentation. For example, if a person crawls on a usual trajectory instead of walking, it shall be detected as an unusual activity (if it is an uncommon activity in that area). This is advantageous over detecting unusual activities based on trajectories alone.

Our framework extends pLSA to classify a novel video as usual or unusual based on the shape and trajectory information found by segmentation. To do so, a new class called *unusual class*, is introduced in addition to the classes found by pLSA during the training phase. Then, a novel video clip is classified into one of these classes. A video is unusual if it is classified to the *unusual class* or if its log-likelihood is below a threshold in the usual class to which it is classified.

In the next section, we discuss the related work. In Sec. 3, an overview of video epitomes is given. Sec. 4.1 gives how clips are generated. In Sec. 4.2 we present our segmentation procedure. Sec. 4.3 discusses the learning procedure that uses pLSA to learn the common paths in the scene and correlations between trajectories. In Sec. 4.4, our extension of pLSA for classifying a novel video clip as usual or unusual is discussed. In Sec. 5, we present the results and conclude in Sec. 6.

2. Related Work

Activity analysis in the context of visual surveillance has become an important area of research recently [1, 14, 12, 15]. The Hidden Markov Models (HMMs) and its variants like parametrized HMM, coupled HMM have been a popular technique for activity analysis [4, 10, 11, 2]. Another popular technique for activity recognition is Bayesian networks [6, 7, 17, 16]. In [6, 7], supervised training using Bayesian formulation is used for estimating the parameters of a multi-layered FSM model that is proposed for activity recognition. Very recently, Bayesian framework has been used for action recognition using ballistic dynamics [18]. This method is based on psycho-kinesiological observations,that is, on the ballistic nature of human movements. These methods are based on feature detection and tracking whereas our method is independent of both these. Since our method uses video epitomes and pLSA we can work with videos with missing frames which leads to loss of trajectory information of the objects.

In statistical methods for activity recognition and unusual activity analysis, the notion that an event is normal is automatically learnt. On this basis, a novel activity is classified as usual or unusual. This is more realistic since in real life scenarios, it is very difficult to predefine all possible usual and unusual activities. Many a times an activity is unusual because there are no previous occurrences of it. Authors in [19], use unsupervised learning for detecting unusual events. We also believe that for a normal event, many similar events should be present in the database. Also, if there are no similar events in the database then the event is unusual. But unlike [19], our method does not extract any features but uses only the foreground patches.

In [9], the aim is to detect irregularities in images and videos. It is posed as an inference process and belief propagation is applied to solve it. Each video is divided into a multitude of space time patches at multiple scales and their descriptors, which capture the local information about the appearance or behavior, are stored in a database. To classify a new observed video, patch descriptors from the query video are compared with the patch descriptors in the database for a similar configuration of patches. If the regions in the query cannot be explained by the database patches, then the query is regarded as suspicious. Authors in [13], apply pLSA model with 'bag of video words' representation for human action recognition. The video words are extracted using space time interest points and do not represent the complete foreground object. They use pLSA to categorize and localize the human actions in a video. Our work is fundamentally different from theirs in that our aim is to learn the usual activity patterns in a scene and not recognize/ characterize human actions. Like [9] and unlike [13], we do not use any features extracted from the video. But unlike [9], we work with foreground space time patches of fixed size, which are extracted using the epitome of the video. This gives us the advantage of working with videos with lost frames. Our method uses pLSA for finding correlations among these patches in space as well as in time. This allows it to detect unusual activities based on both the shape and the trajectory as well as learn the correlations between the landmarks in the scene. Therefore, our method is capable of detecting irregularities in behavior and learning the common paths and shapes in the scene. Moreover, we extend pLSA for classifying a novel video as containing usual or unusual activities.

3. Background

Recently, video epitomes were introduced by Jojic and Frey [3]. We give a brief introduction to video epitomes for the sake of completeness.

3.1. Video Epitome

A video epitome is a compact representation of a video which is smaller in both space and time as compared to the input video. The epitome is learnt from a large collection of 3D patches (*i.e. patches in both space and time*) from the input video and captures nearly all spatial and temporal patterns in the video. These epitomes have been used for various reconstruction tasks such as dropped frame recovery, video in-painting, video super-resolution, etc. An input video can be considered as a 3D array $v_{x,y,t}$ of real valued pixels where $x \in \{1, ..., X\}$, $y \in \{1, ..., Y\}$ and $t \in \{1, ..., T\}$. On the other hand, it's epitome is of size $X_e \times Y_e \times T_e$ where $X_e Y_e Z_e < XYT$ and each pixel in the epitome corresponds to a Gaussian probability distribution which is parametrized by a mean $\mu_{x,y,t}$ and a variance

$\phi_{x,y,t}$.

$$e_{x,y,t}(.) = N(.; \mu_{x,y,t}, \phi_{x,y,t}) \quad (1)$$

A particular pixel value v can be evaluated under any of these probability distributions. To build the epitome, first sample 3D patches are drawn from the input video and then a learning algorithm is applied to construct the epitome. Each 3D patch describes a set of coordinates S. For example, a patch starting at location $(5, 8, 7)$, (where 7 is the frame number $(5, 8)$ are x and y coordinates) of size say, $20 \times 20 \times 4$ describes the set of coordinates $S = \{5, ..., 24\} \times \{8, ..., 27\} \times \{7, ..., 10\}$. Here $S(k)$ denotes the k^{th} coordinate in the patch, for example, $S(1) = (5, 8, 7)$.

Let v_S denote the observed pixel values in the video cube described by the set of coordinates S in the input video and c_S denote the video cube predicted by the epitome for same coordinates S in input video. The goal of the learning algorithm is to make the predicted cubes similar to original cubes i.e. $c_S \approx v_S$. The cube c_S is predicted using a set of distributions in the epitome. Let e_T denote the set of distributions from epitome e at coordinates T. Assuming that T and S are of the same size, the probability density evaluated using distributions at coordinates T and predicted values c_S is $e_T(c_S)$.

$$p(c_S|T) = e_T(c_S) = \prod_{k=1}^{|T|} e_T(k)c_S(k) \quad (2)$$

The above equation describes the probability model for an individual cube. However, for overlapping coordinates S and S' in the video, predicted video cubes c_S and $c_{S'}$ should make similar predictions for overlapping pixels. We now explain how the whole video is generated from the epitome. Consider a video pixel $v_{x,y,t}$ represented as a dot in Fig. 1. The overlapping video cubes in the input video containing (x, y, t) are $\{S_{i_1}, \ldots, S_{i_n}\}$, two of which are shown in the input video. These predicted cubes,$\{c_{S_{i_1}}, \ldots, c_{S_{i_n}}\}$ are generated from the epitome by randomly selecting locations in the epitomes $\{T_{i_1}, \ldots, T_{i_n}\}$, represented by dots in the video epitome, and then generating the pixel values for the predicted cubes under the distribution given by the epitome for the locations, $e_{T_{i_1}}(c_{S_{i_1}}), \ldots, e_{T_{i_n}}(c_{S_{i_n}})$. Each of these cubes $c_{S_{i_1}}$, make predictions for the pixel $v_{x,y,t}$ and these predictions are combined to generate a single pixel value. We discuss in Section4.2 how we use video epitomes for segmenting the foreground from the background.

4. Activity modeling using pLSA

In order to learn correlations between foreground epitome patches (words) and build models for usual activity patches using pLSA, we need to define *words* and *documents*.

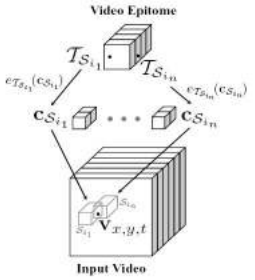

Figure 1. Reconstruction of video from its epitome [3]

Figure 2. Result of segmentation performed using video epitomes.

4.1. Clip Generation

To generate a document, we divide a video into short clips of d frames each. Since video epitomes are used for the purpose of segmentation, dropping frames while creating these documents does not effect the result. Therefore, the clips can also be generated by dropping frames and taking alternate or every third frame from a longer clip such that every document is of length d frames.

4.2. Segmentation

We use video epitomes for segmenting the foreground objects from the background. Suppose E_b is the epitome of a background clip V_b and E_a is the epitome of a clip V_a having some activity. Assume that the epitomes are of size $X_e \times Y_e \times T_e$ and were constructed using patches of size $P_x \times P_y \times P_z$. The premise of our segmentation procedure

is that if a 3D patch of E_a predicts a background 3D patch of V_a then it is likely to have a high correlation with one of the 3D patches of E_b. Similarly, if a 3D patch of E_a predicts a foreground patch of V_a then it will have low correlation with all the possible patches of E_b since patches of E_b can only predict the background. Therefore, foreground epitome patches can be computed from epitome E_a and E_b by using an appropriate threshold on the correlation measure between patches from the two epitomes. Since epitome patches are randomly placed, a patch from E_a would have to be correlated with each patch of E_b in a sliding window fashion. To do this efficiently, first we reconstruct the videos V_a and V_b from epitomes E_a and E_b respectively. After reconstruction, we hash the epitome patches P_{ea} that predict a patch at position T in V_a and similarly, for epitome patches P_{eb} that predict the patch at position T in V_b. Then, correlation is found between patches from E_a and E_b that predict a video cube at the same position T and a threshold is put over the correlations obtained to get the foreground patch locations. Since we use a common threshold for all video clips, it is possible that some of the foreground patches remain undetected. We show in the next section that using pLSA we are able to discover the undetected foreground patches.

4.3. Learning spatio-temporal correlations through pLSA

We use pLSA to obtain spatio-temporal correlations among the foreground patches. This overcomes the two main issues of inaccurate segmentation and broken tracks. Since segmentation is performed at a coarse level, some foreground patches remain undetected. Spatial correlations among the patches from all the clips in the training data helps in correlating patches that belong to the same object but are not connected to each other because of missing patches. The spatio-temporal correlations among the patches give the approximate shape and trajectory of the objects. pLSA performs spatio-temporal clustering of these space-time patches and these clusters represent the usual activity patterns in the scene. We now explain the learning procedure.

Document Generation: The documents are the clips which are generated as described in Sec. 4.1. We assume that overlapping patches denote one object and if there are more than one segmented object in a clip, then we create a separate document for each of them.

Codebook generation: We find the locations of the space-time patches in the original video clip by reconstructing the video from its epitome. Let P_i denote the collection of foreground patches obtained for clip i. Then, each P_i is a $N_i \times 3$ matrix where N_i is the number of foreground patches in clip i. Each row (x_j, y_j, t_j) corresponds to the location of the j^{th} foreground patch in the reconstructed video. The vocabulary is built by clustering on the starting location (x, y, t) of all the foreground patches across all the video clips. This collection of patches, C is an $N \times 3$ matrix where N is the total number of foreground patches taken over all video clips. Each cluster obtained is represented as a word as shown in Fig. 3.

Learning process: We now explain how this clustering

Figure 3. The black patches in the images are the sample words from the vocabulary.

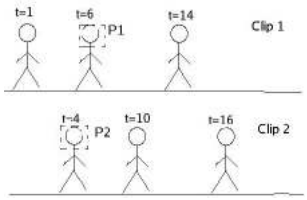

Figure 4. An example situation

over patches and pLSA helps us in learning the shape and paths taken by the people in the scene. The documents in this case are the video clips and the foreground patches in it form the words in the document. Suppose the scene is as described in Fig. 4 where people walk from left to right. The two clips show the various positions of people at different times. But none of the clips can alone describe the complete path taken. This situation is similar to the tracking problem where it is difficult to obtain the complete trajectories. pLSA can discover the complete path through these small clips because both these clips will have some common words. For example, as indicated in Fig. 4, patch P_1 and P_2 are likely to be in one cluster and thus, represent the same word across those two clips. Many more words will be common due to patches obtained on the arms and legs and hence, the two documents are likely to be classified into the same class. The co-occurrence matrix to be used by pLSA is built as follows :-

For all patches in **C**

- Identify the document d of the patch
- Identify the word (or *cluster*) w to which the patch belongs
- Update $n(d, w) = n(d, w) + 1$;

Through pLSA, we are able to discover the various hidden classes z which capture motion of the foreground patches as well as their spatial arrangement. Hence, each of the video clips can be classified into one of the hidden classes using the standard *Expectation Maximization* algorithm where the E and the M step are as given below: *E-step*

$$\mathbf{P}(z|w,d) = \frac{\mathbf{P}(w|z)\mathbf{P}(z|d)}{\sum_{z\in\mathcal{Z}}\mathbf{P}(w|z)\mathbf{P}(z|d)} \tag{3}$$

M-step

$$\mathbf{P}(w|z) = \frac{\sum_{d\in\mathcal{D}} n(d,w)\mathbf{P}(z|w,d)}{\sum_{w\in\mathcal{W}}\sum_{d\in\mathcal{D}} n(d,w)\mathbf{P}(z|w,d)} \tag{4}$$

$$\mathbf{P}(z|d) = \frac{\sum_{w\in\mathcal{W}} n(d,w)\mathbf{P}(z|w,d)}{\sum_{z\in\mathcal{Z}}\sum_{w\in\mathcal{W}} n(d,w)\mathbf{P}(z|w,d)} \tag{5}$$

Here, $P(z|d)$ is the probability of a topic z occurring in document d, $P(w|z)$ is the probability of a word w occurring in a topic z and $P(z|w,d)$ is the probability of a topic z given the word w in document d. After convergence, we assign a topic to the words as follows:

$$topic(w) = argmax_{z\in Z}\{P(w|z)\} \tag{6}$$

Each collection of words $C_i = \{w|topic(w) = i\}$ gives us the correlation between the different landmarks in the scene.

Shape recovery using pLSA: As mentioned before, due to a threshold on the correlation measure, sometimes we are not able to get all the foreground patches in some of the clips. Assume that for some of the clips, say a collection $S_{correct}$, we have obtained all the foreground patches. We recover lost foreground patches for a clip C with the help of $S_{correct}$. This can be done because through pLSA we learn the spatial arrangement of patches on an object using $S_{correct}$. We now describe how these foreground patches can be recovered for a clip C using the probabilities computed as above. Assume that a clip C for which we want to recover lost foreground patches, gets classified to class z. We pick a clip C' from class z which has the maximum probability $P(z|d)$ and look at the spatial arrangement of patches in a particular frame of C'. Suppose, at frame i in the clip C' there exists a collection of 2D foreground patches P. We compute the centroid c of P and relative position of patches R_i from the centroid as follows:

$$c = (1/|P|)(\sum_i x_i, \sum_i y_i) \tag{7}$$

$$R_i = (x_i - c_x, y_i - c_y) \tag{8}$$

Suppose now we want to recover lost foreground patches for clip C. For each frame in C we compute the centroid c_C of foreground patches in that frame as explained above.

Figure 5. Recovered foreground patches using pLSA.

Then, we generate a new collection of patches N_i for this frame as follows:

$$N_i = R_i + c_C \tag{9}$$

These generated patches N_i could possibly be the foreground patches for the particular frame in C. We declare a generated patch P_{gen} to be a foreground patch using background subtraction, that is, if the following holds:

$$|P_{gen} - P_{back}| > \epsilon \tag{10}$$

where P_{back} is the background epitome patch corresponding to the coordinates described by P_{gen} and ϵ is an appropriate threshold. In Fig. 5, the left column shows that sometimes some foreground patches are not recovered during segmentation. The right column shows the corresponding frames with the foreground patches recovered using pLSA.

4.4. Novel video classification

After we have learnt the probabilities as discussed above, we use them to classify a novel video as usual or unusual. An unusual video clip which cannot be explained by our training set and therefore, is marked as suspicious. In order to classify a novel video or a collection of novel videos, we first segment out the foreground as explained earlier and then build a co-occurrence matrix as follows:
For each patch P in the novel video

- Identify the word w closest to the patch.
- If $d(P,w) > r(w)$ then $n(d,w_u) = n(d,w_u) + 1$
- Otherwise, $n(d,w) = n(d,w) + 1$

where, $d(P, w)$ denotes the Euclidean distance between the location of patch P and cluster center w, $r(w)$ denotes the radius of the cluster w and w_u represents an unusual word that has not been seen in training. Since w_u is not in our training vocabulary, we add this word and update the probability $P(w_u, z)$ as 0 for all z. Also, we add one more hidden class z_u which represents an unusual class such that $P(w, z_u) = 0$ and $P(w_u, z_u) = 1$.

We now apply the EM algorithm again, except that we do not update $P(w|z)$ but use the values computed from the learning process. If a novel video gets classified into the unusual class z_u, then it is marked as suspicious. However, this is not the only case when a novel video is marked suspicious. A novel video can be unusual even if it does not contain unusual words. Hence, we also mark a test video as suspicious if it's likelihood is low compared to the likelihood of the training documents. An example of such a document is provided in the next section. Log-likelihood of a document d is computed from the estimated probabilities $P(w|z)$ and $P(z|d)$ as follows:

$$LL(d) = \sum_{w \in \mathcal{W}} \mathcal{N}(d, w) \log(\sum_{z \in \mathcal{Z}} P(w|z)P(z|d)) \quad (11)$$

Low likelihood implies that the document cannot be explained by the training data with much confidence.

5. Results

We conducted experiments in both outdoor and indoor scenes. We have used the video epitome code available at [8] for learning the epitome and reconstructing the video clip from its epitome.

5.1. Results: Indoor scene

In the indoor scenario, we collected video data of over four days and made 49 clips of size $320 \times 240 \times 20$ as the training data. The size of the epitomes used is $45 \times 50 \times 6$. These got classified to 4 hidden classes using pLSA. Each of these classes correspond to the structure of patches and the path taken by the objects. A few of the frames from the different documents belonging to the hidden class 2 are shown in the Fig. 6. Class 2 contains documents in all of which people walk in the center of the corridor away from the camera. Fig. 7 shows a few frames from class 3. In these documents people walk along the right wall of the corridor towards the camera. This shows that pLSA learns the different classes based on the position of the space time patches. As shown in Fig. 8, we have been able to obtain the correlations between various landmarks in the scene. Red point denotes the starting point of a trajectory while the blue point denotes the endpoint. Each trajectory in Fig. 8(a) represents

Figure 6. Frames of documents in class 2 of the indoor scene. Frames in a row are from the same document.

Figure 7. Frames of documents belonging to class 3. Frames in a row are from the same document.

a clip or document belonging to a particular class. These trajectories are obtained by computing the centroid of the foreground patches on a frame by frame basis. No feature tracking has been performed here. In both Fig. 8(a) and (b), none of the trajectories represent a complete path taken by the people but the correlations between these broken trajectories show the complete paths that are commonly walked across in the corridor. They also show that fixed sized clips that are randomly generated can be used for the learning process and it is not required that a clip should contain the complete path taken by a person.

Fig. 9 shows frames from the unusual activity clips which are provided as the novel test videos to our system. The video corresponding to activity in Fig 9(a) gets classified to the unusual class since it contains too many unusual words. These words are unusual because no patches occur at the height of the ceiling in the training clips. The video clip corresponding to Fig. 9(c) is unusual because crawling is not a usual activity as depicted by its low log-likelihood. The clip corresponding to Fig. 9(d) is unusual because in general, people do not walks along the width of the corridor.

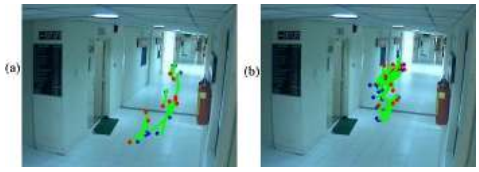

Figure 8. Correlations obtained for two different classes.

Figure 9. Examples of unusual activities in the indoor scene.

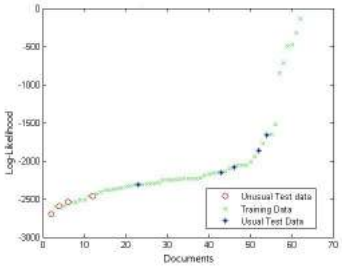

Figure 10. Plot of the log-likelihoods of the documents in the indoor experiment

In the plot shown in Fig. 10, shows the log-likelihood of the documents. The green crosses are the log-likelihood of the training data while the red circles at the bottom of the curve correspond to the unusual clips shown in Figs. 9(b), (c) and (d). The blue stars ('*') are the usual test cases that

Figure 11. Frames from documents in class 2 for the outdoor scene. Each row corresponds to frames from the same document.

Figure 12. Frames from documents in class 5 for the outdoor scene. Each row corresponds to frames from the same document.

were supplied to the detection system along with unusual clips. They have been correctly classified as usual activities. The runtime is 30 minutes for epitome generation for each clip on a dual core 4GB RAM machine and 1 minute for the learning and classification on a dual core 1GB RAM machine.

5.2. Results: Outdoor scene

Figure 13. Correlations between various landmarks in the scene as described by the different classes.

In the experiment in an outdoor scene, the training data consisted of 75 clips of size $320 \times 240 \times 20$ that were randomly generated. Here, too the size of the epitomes used

is $45 \times 50 \times 6$. These got classified to 6 hidden classes through pLSA. Fig. 11 shows a few frames from the documents that were classified into class 2. Documents that got classified into class 5 are shown in Fig. 12. Even when there are multiple objects in the same clip, the corresponding documents contain only one segmented object each. Fig. 13 shows the correlations between different landmarks in the scene that have been captured by using pLSA. Some unusual activities in the scene are shown in Fig. 14. The clip corresponding to Fig. 14(a) is classified to the unusual class while the others are classified as unusual due to their low log-likelihood measure. In clip of Fig. 14(b) the person is taking an unusual path whereas in clip of Fig. 14(c) the person is crawling, which is an unusual activity in this scene. Fig. 15 gives the plot of the log likelihood of the documents.

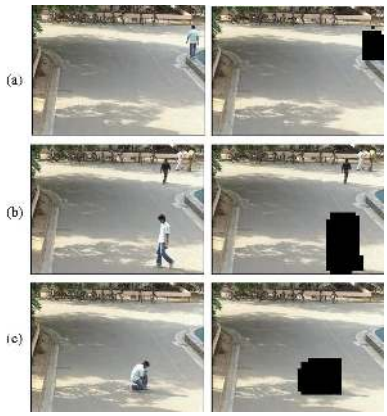

Figure 14. Unusual activities in the outdoor environment.

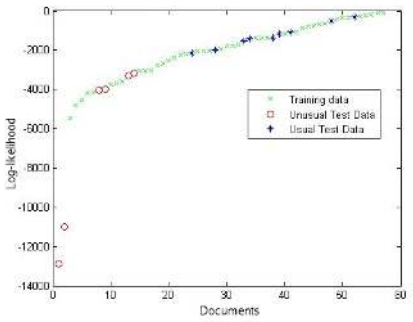

Figure 15. Plot of log-likelihood of the documents in the outdoor experiment

The green crosses correspond to the log-likelihood of the training clips, the red circles correspond to the unusual activities including those in Fig. 14(b) and (c) and the blue stars correspond to the test cases which depict usual activities. The runtime for epitome generation of each clip is the same as before and is 2.5 minutes for the learning and classification on a dual core 1GB RAM machine.

6. Conclusion

We see that video epitomes can be used for segmenting the foreground objects in the scene in the form of space time patches. The correlations among these space time patches define the approximate shape and trajectory of the object. We have used pLSA to find the spatio-temporal correlations among the patches. This has allowed us to learn the usual activity patterns on the basis of both shape and trajectory of the objects. Our framework finds the correlations among the landmarks in the scene as well as finds the common paths taken in the scene, using pLSA. For classifying a novel video clip as usual or unusual, we have presented an extension of pLSA.

References

[1] M. F. Abdelkader, A. K. Roy-Chowdhury, R. Chellappa, and U. Akdemir. Activity representation using 3d shape models. *EURASIP Journal on Applied Signal Processing*, 2008.

[2] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. *In Proc. CVPR*, pages 994–999, 1997.

[3] V. Cheung, B. J. Frey, and N. Jojic. Video epitomes. *In Proc. CVPR*, 2005.

[4] N. P. Cuntoor, B. Yegnanarayana, and R. Chellapa. Activity modeling using event probability sequences. *In IEEE Trans. Image Processing*, 17(4):594–607, April 2008.

[5] T. Hoffmann. Probabilistic latent semantic analysis. *In SIGIR*, pages 50–57, 1999.

[6] S. Hongeng, F. Bremond, and R. Nevatia. Representation and optimal recognition of human activities. *In Proc. CVPR*, pages 1818–1825, 2000.

[7] S. Hongeng and R. Nevatia. Multi-agent event recognition. *In Proc. ICCV*, pages 84–93, 2001.

[8] http://www.psi.toronto.edu/ vincent/sourcecode.html.

[9] M. Irani and O. Boiman. Detecting irregularities in images and videos. *In Proc. ICCV*, 2005.

[10] V. Kettnaker. Time-dependent HMMs for visual intrusion detection. *In IEEE Workshop on Event Mining: Detection and Recognition of Events in Video*, 2003.

[11] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia. Event detection and analysis from video stream. *IEEE Trans. on PAMI, 23(8)*, pages 873–889, 2001.

[12] V. Nair and J. J. Clark. Automated visual surveillance using hidden Markov models. *In Proc. ICCV*, pages 88–93, 2002.

[13] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *In Proc. BMVC*, 2006.

[14] W. Niu, J. Long, D. Han, and Y.-F. Wang. Human activity detection and recognition for video surveillance. *In Proc. ICME*, 2004.

[15] N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. *In Fourth IEEE Int. Conf. on Multimodal Interfaces*, pages 3–8, 2002.

[16] N. Oliver, B. Rosario, and A. Pentland. A bayesian computer vision system for modeling human interactions. *In Proc. ICVS*, pages 255–272, 1999.

[17] S. Park and J. K. Aggarwal. A hierarchical Bayesian network for event recognition of human actions and interactions. *In Association for Computing Machinery Multimedia Systems Journal*, 2004.

[18] S. N. Vitaladevuni, V. Kellokumpu, and L. S. Davis. Action recognition using ballistic dynamics. *In Proc. CVPR*, 2008.

[19] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. *In Proc. CVPR*, pages 819–826, 2004.