

Unsupervised Real-Time Unusual Behavior Detection for Biometric-Assisted Visual Surveillance

Tsz-Ho Yu and Yiu-Sang Moon

The Chinese University of Hong Kong

Abstract. This paper presents a novel unusual behaviors detection algorithm to acquire biometric data for intelligent surveillance in real-time. Our work aims to design a completely unsupervised method for detecting unusual behaviors without using any explicit training dataset. To this end, the proposed approach learns from the behaviors recorded in the history; such that the definition of unusual behavior is modeled according to previous observations, but not a manually labeled dataset. To implement this, pyramidal *Lucas-Kanade algorithm* is employed to estimate the optical flow between consecutive frames, the results are encoded into flow histograms. Leveraging the correlations between the flow histograms, unusual actions can be detected by applying *principal component analysis* (PCA). This approach is evaluated under both indoor and outdoor surveillance scenarios. It shows promising results that our detection algorithm is able to discover unusual behaviors and adapt to changes in behavioral pattern automatically.

1 Introduction

Visual surveillance is ubiquitous in modern societies: *closed-circuit televisions* (CCTV) and other visual surveillance systems are common in public places like airports, train stations and banks. Traditional visual surveillance systems have provided valuable forensic evidence in crime and terror investigations, yet they are not as effective for crime prevention. Because of the rapid growth of surveillance cameras installed and inadequate manpower to supervise them, discovering suspicious activities manually from the surveillance video data becomes inefficient. This has led to an urgent demand for automatic methods to analyze vast quantities of surveillance videos.

Therefore, the fusion of biometric technologies and visual surveillance has attracted much attention recently. Several *biometric-assisted* visual surveillance systems has been proposed, identifying people using gait or face recognition. Besides recording videos for forensic investigations, biometric-assisted surveillance are designed to detect unusual events, identify the people involved and give corresponding warning immediately. For instance, Zhou et al. presented a master-slave camera network to acquire biometric imagery [1]; Nishiyama et al.



Fig. 1. CCTV images from London bombing on 7 July 2005

designed a surveillance system which recognizes faces in a cluster of moving people [2], and Chan et al.[3] used a Pan-Tilt-Zoom (PTZ) camera to capture face images in a corridor for recognition.

Figure 1 illustrates the motivation for detecting unusual behavior, and the importance of applying biometrics in visual surveillance. This is an excerpt of the CCTV footages recorded during the London bombing terrorist attack on 7 July 2005. Interestingly, the suspects were actually monitored by security cameras, however it still took months for police to examine the 2500 tapes of surveillance videos [4,5]. If the surveillance system could detect the unusual behaviors of the suspects, they would be identified in advance, number of victims could have been minimized.

Unfortunately, due to the inevitable mechanical delays in PTZ cameras and other computational limitations, every biometric-assisted system can only handle a certain amount of people simultaneously. Hence, it is technically infeasible to identify every person using biometrics. To tackle this problem, people are monitored selectively — we are just interested in keeping an eye on the potential troublemakers who behave abnormally. Once an unusual activity is detected, biometric features of the target person is captured and identified [3]. Being the first step for biometric-assisted surveillance, unusual behavior detection directly determines the reliability of the whole system.

In this paper, we present a novel algorithm to detect unusual behaviors for acquiring biometric data in surveillance videos. Leveraging the large correlation between normal behaviors, this technique is specialized for a deployable real-time surveillance system. Utilizing optical flow for motion segmentation and eigenspace transformation for classification, our proposed algorithm is able to detect unusual behavior in real-time without providing any training data. Hence, the surveillance system is able to identify the misbehaving person immediately by means of face recognition or gait analysis.

The rest of the paper is structured as follows: Section 2 reviews the related researches and highlights the contribution of our work. Section 3 describes the feature extraction techniques using pyramidal optical flow from surveillance videos [6,7]. Section 4 explains the unsupervised detection algorithm in order to acquire biometrics data for identity verification. Experiments have been performed to justify our proposed method in various real life scenarios, evaluation results are reported in Section 5. Finally, we draw the conclusion in Section 6.

2 Detecting Unusual Behavior in Video

Techniques for unusual behavior detection has been considered as a challenging topic in computer vision for years. For detecting irregularities in surveillance

video, Xiang et al. developed a sophisticated event-based technique for classifying unusual behavior in video footages incrementally [8,9]. Zhong et al. took a different perspective on detecting unusual activity in videos, motivated by a similar problem in document keyword analysis [10]. Other approaches for detecting unusual behaviors include optical flow [11], silhouette contours [12] or Bayesian inference [13].

Although the importance of behavioral analysis in visual surveillance system has long been recognized, some issues remain largely unsolved under realistic and uncontrolled circumstances. First, it is difficult to give an explicit definition for unusual behaviors, even it is very easy to be recognized manually. A suspicious behavior in one scene can be regarded as normal in other environments. Currently, much work relies on a supervised learning approach, the “unusual behaviors” are required to be well-defined and labeled in the dataset for classifier training. These approaches is not feasible in real surveillance, as the definition of “unusual behaviors” changes over time. In addition, some of detection algorithms rely on detection of local features [12,8,9]. These approaches can achieve very high accuracy when the background is well modeled. Still, they require accurate segmentation of body silhouettes, which can be affected by occlusions or clutter background. Moreover, many techniques on unusual behavior detection cannot be performed in real-time, as they require the complete video for normal/abnormal action modeling [10,12,13].

The main contribution of our work is to present a simple unsupervised algorithm for detecting unusual behavior in real-time, such that the target can be identified immediately using biometric features. Instead of using an explicit training set with well-defined models of “abnormal behaviors”, an unsupervised approach is utilized to learn previously-recorded behaviors as prototypes. No training data is required as the proposed algorithm adapts to new behavior pattern dynamically. By learning live samples captured from the actual environment, the definition of unusual behavior can be adjusted incrementally, so the classification criteria gradually changes adaptively. This algorithm is effective yet inexpensive to be performed in real-time, making promptly response to capture biometric features like face images or gait animations. Features of behaviors are extracted using optical flow field, then the recorded behavior samples are transformed into eigenspace by PCA.

3 Feature Extraction and Representation

Framework for the proposed unusual behavior detection is illustrated in Figure 2. For each frame recorded from the visual surveillance system, moving objects are

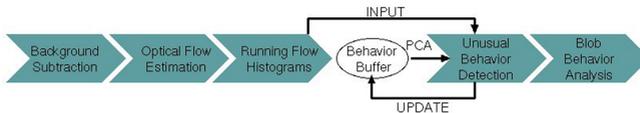


Fig. 2. Framework of the proposed detection algorithm

extracted by using background subtraction. Pyramidal optical flow is applied on the movie objects to analyze their motion direction and velocity. The overall movements in the scene are represented in the running flow histograms which will be discussed later in this section. The histograms samples are stored and transformed using PCA; unusual behaviors in the videos is then detected and located using previous observations.

3.1 Preprocessing

Surveillance videos captured from security cameras are available in gray-scale image sequences. At a specific time, the surveillance system has captured a video segment consisting T frames. This video segment \mathcal{I} can be represented as a series of integer matrices: $\mathcal{I} = \{I_0, I_1, I_2, \dots, I_i, \dots, I_T\}$; where I_n is the n th frame in the video, and $I_n(x, y)$ represent the *intensity* at frame coordinate (x, y) . Before extracting moving objects from the frames, it is necessary to classify stationary frames and frames with actions, this is done by computing the image difference between two consecutive frames:

$$Id_i(x, y) = \begin{cases} 1 & \text{when } I_i(x, y) - I_{i-1}(x, y) \leq thresh_{m1} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$Id_{i,sum} = \sum_x \sum_y Id_i(x, y) \quad (2)$$

$$\text{Frame } I_i \text{ is stationary when } Id_{i,sum} \leq thresh_{m2}. \quad (3)$$

The thresholds $thresh_{m1}$ and $thresh_{m2}$ can be adjusted according to the surveillance environment, they determines the sensitivity of the motion detection. Motionless frames are discarded from detection, assuming that no behavior is captured in the frames.

Foreground pixels are subtracted from the background by employing *mixture of Gaussian* method proposed by Stauffer and Grimson [14]. Figure 3 shows the foreground regions obtained from a frame. It is interesting to note that the foreground region subtracted is not always as precise as expected. In spite of that, our approach can still tolerate with inaccuracies in background subtraction, because the shapes of foreground regions will not be treated as a feature for detection.

3.2 Running Flow Histograms

Instead of measuring the trajectories of moving objects [12] or motion-history image (MHI) [15], optical flow method is chosen because only two consecutive frames are required to estimate the movement fields in the scene, which enables faster responses for a real-time surveillance system. In addition, optical flow methods is capable of finding different motion vectors within one object, offering more information for unusual behavior detection.



Fig. 3. Results of background subtraction using mixture of Gaussian method: (a) original frame; (b) foreground regions of frame (a); (c) original frame; (d) foreground regions of frame (c)

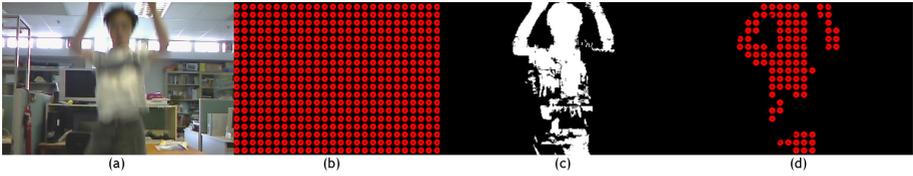


Fig. 4. Number of feature points are reduced using the foreground pixels as mask: (a) original frame; (b) positions of original feature points; (c) the foreground mask; (d) remaining feature points after masking

To ensure that the features points are evening distributed, a dense-feature optical flow field estimation is employed to compute the movements of foreground objects. The amount of feature points are reduced by using a grid mask, which is illustrated in Figure 4, to prune away the inaccurate flows caused by shaking backgrounds. Assuming the movements are small enough, the deformation between two consecutive frames can be estimated using the Lucas-Kanade algorithm.

Addressing the problem of detecting fast movements by Lucas-Kanade algorithm, the flow estimation is augmented by constructing an image pyramid [7]; Figure 5 shows the result of the optical flow algorithm. In an array of flows F_n in frame I_n , each feature point is represented as a four-dimensional feature vector f_n :

$$\begin{aligned} F_i &= \{f_{i1}, f_{i2}, \dots, f_{ij}, \dots, f_{iJ}\} \\ f_{ij} &= [X_{ij}, Y_{ij}, dx_{ij}, dy_{ij}], \end{aligned} \tag{4}$$

where (dx_{nj}, dy_{nj}) and (X_{nj}, Y_{nj}) denote the velocity and location of the feature respectively. For each individual frame with motion detected, a flow histogram $h'_i(d, v)$ is constructed, with $N_d \times N_v$ bins in direction and velocity respectively, as described in 5 and 6.

$$\begin{aligned} \theta_{ij} &= \tan^{-1}\left(\frac{dy_{ij}}{dx_{ij}}\right) \\ S_{ij} &= \sqrt{(dy_{ij})^2 + (dx_{ij})^2} \end{aligned} \tag{5}$$

$$h'_i(d, v) = \text{no. of vector that satisfy } \begin{cases} d\theta * (d - 1) \leq \theta_{ij} \leq d\theta * d \\ ds * (v - 1) \leq S_{ij} \leq ds * v \end{cases} \tag{6}$$



Fig. 5. Optical flow estimations using pyramidal Lucas-Kanade algorithm

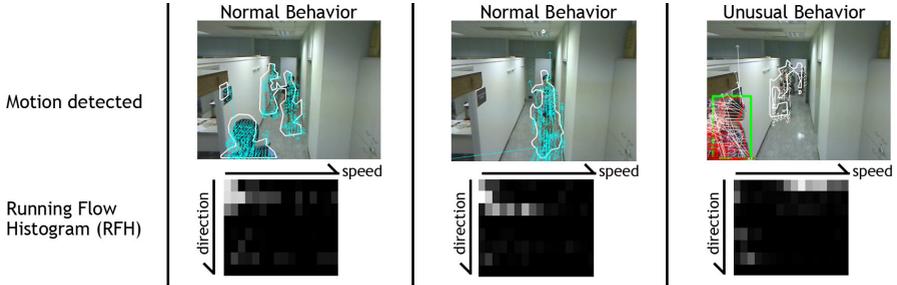


Fig. 6. The *Running Flow Histograms* computed from both normal and unusual behaviors. Note that the histograms of normal behaviors look similar.

For smoothing the transition and balancing the bin values between flow histograms, *running flow histograms (RFH)* are computed on-the-fly in 8.

$$h_i''(d, v) = \alpha_h \times \text{Normalize}(h_i'(d, v)) + (1 - \alpha_h) \times h_{i-1}(d, v) \quad (7)$$

$$h_i(d * N_v + v) = h_i''(d, v). \quad (8)$$

It is a running average of normalized histograms along the time-line, the parameter α_h determines how much the histograms are smoothed with the previous histogram. Figure 6 visualizes the RFHs computed from different behaviors. In order to perform PCA in the detection stage, every RFH computed is reshaped to a column vector with size $N_d N_v \times 1$.

4 Unsupervised Detection Algorithm

The key concept of the proposed method is that normal behaviors possess high correlation between each others. Therefore, unusual behaviors can be detected by comparing with normal behaviors recorded in the history. If the current behavior presents a large deviation from the observed behaviors in the history, it is classified as unusual. To implement this, a buffer of behavioral data is kept in the system to model the normal behaviors dynamically.

4.1 Eigenspace Transformation

As described in Figure 2, a behavior buffer is kept in the program to record samples of RFHs observed previously. When a new i th frame is acquired from a

surveillance camera, the corresponding RFH h_i is computed. On the other hand, a set of eigenvectors and eigenvalues are obtained by applying PCA on the behavior buffer. Assume that there are S_b RFHs stored in the behavior buffer, the RFHs are concatenated to form a $S_b \times N_d * N_v$ matrix: $H^* = [h_1^* h_2^* h_3^* \dots h_{S_b}^*]$. The matrix of eigenvectors V and diagonal matrix of eigenvalues E can be calculated by solving 9:

$$V^{-1}Cov(H^*)V = E. \quad (9)$$

The eigenvectors are sorted by their corresponding eigenvalues in descending order, the first N_e eigenvectors sorted are kept in a matrix V' for unusual behavior detection.

4.2 Detecting Unusual Behavior

Once matrix V' is obtained, the current RFH is projected onto eigenspace and back-projected immediately in 11:

$$\text{Projection: } \mathbf{h}_i = (h_i - \overline{H^*}) * V'^T \quad (10)$$

$$\text{Back-projection: } h_i^r = \mathbf{h}_i * V' + \overline{H^*}, \quad (11)$$

where $\overline{H^*}$ is the element-wise average vector of H^* . As an unusual behavior do not have high correlation with normal behaviors, its RFH cannot be back-projected completely using the eigenvectors computed from normal behaviors. Leveraging this property, the recovery rate R determines if there is unusual behavior in the current frame:

$$R = 1 - \frac{\|h_i^r - h_i\|}{\|h_i\|}. \quad (12)$$

A frame is classified as “unusual” when the recovery rate R is smaller than a pre-defined threshold $thresh_r$. This threshold can be adjusted to control the sensitivity of the detection algorithm:

The frame I_i contains unusual behavior(s) if its recovery rate $R_i \leq thresh_r$ (13)

4.3 Locating Objects of Unusual Behavior

The location of the unusual behavior can be reconstructed using the difference between the original and back-projected RFH:

$$W_i = |h_i^r - h_i|. \quad (14)$$

The vector W represents how “unusual” a flow vector is in the current frame, the difference between two RFHs represents the flow vectors that cannot be recovered by PCA. A “anomaly weight” $W_i(d * N_v + v)$ is assigned to all flow vectors in the scene, according to its corresponding bin in h_i . For each connected component B extracted from the foreground, the average “anomaly” of

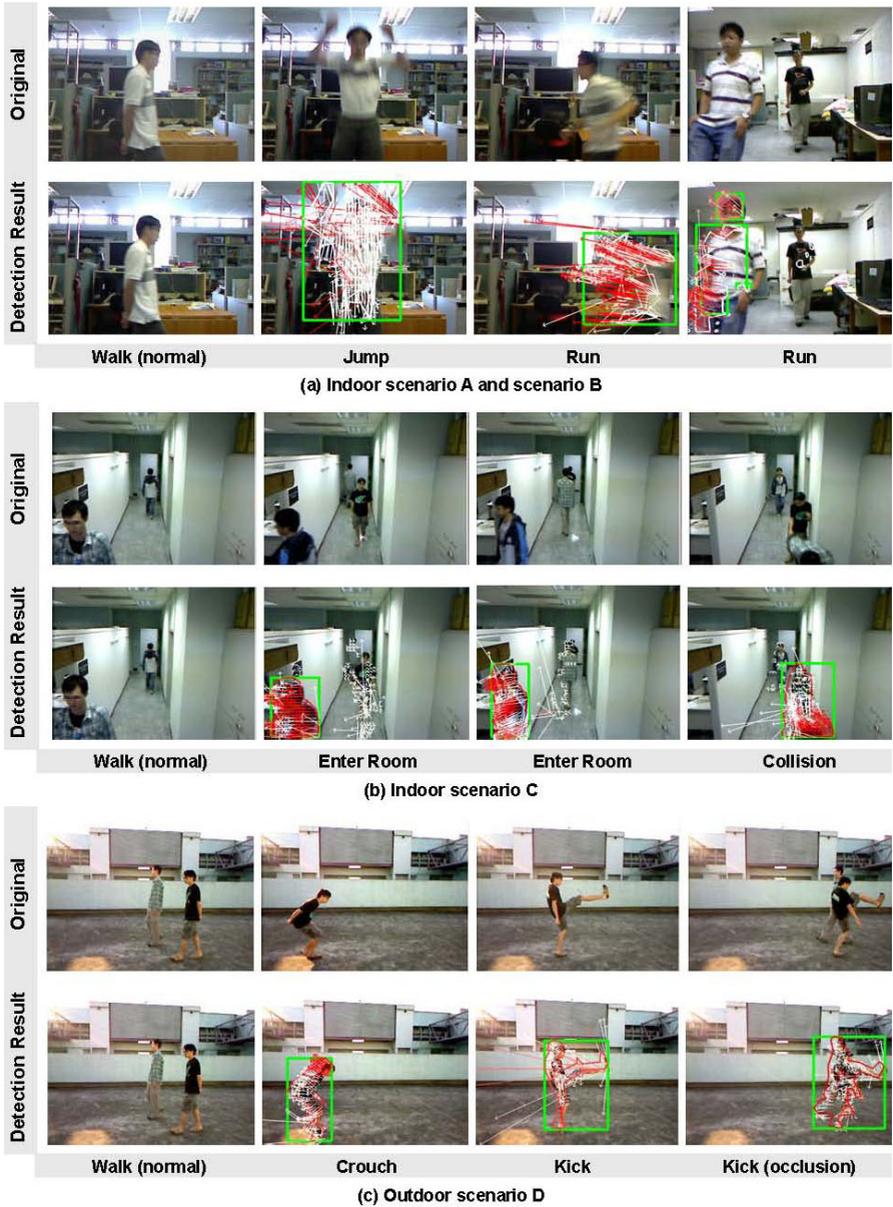


Fig. 7. Sample Detection Result: (a) Detection results in scenario *A* and *B*; (b) Detection results in scenario *C*; (c) Detection results in scenario *D*

a foreground region can be calculated by summation of all weights inside the connected component divided by its own area:

$$A(B_{ik}) = \frac{\sum_{F_i \in B_{ik}} W_i}{area(B_{ik})}. \tag{15}$$

Consequently, each connected component detected in the scene has been assigned an average “anomaly”. This value indicates how the object’s behaviors is differ from the normal behaviors in the history. A blob B_{ik} can be regarded as abnormal if $A(B_{ik}) \geq thresh_r \cdot 2$.

4.4 Behavior Model Update Scheme

In the unusual behavior detection algorithm, a behavior buffer is used as the prototype of normal behaviors. In real situations, the behavioral pattern changes gradually over time, therefore the behavior buffer needs to be updated continuously with new behaviors.

In our work, the behavior buffer is implemented using a circular linked list with size S'_b where $S'_b \leq S_b$. The behavior buffer employs a “first-in, first-out” design: when a new frame enters the system, it will be evaluated by its recovery rate R ; if R is greater than the replacement threshold $thresh_h$, the RFH of the new frame will replace the oldest frame inside the behavior buffer. During the detection process, only the oldest S_b entries are selected to compute the eigenvectors. The most recent frames are not used in detection because they can achieve very high correlations with the current frame, over-weighing the effect of unusual behaviors. As a result, the behavior buffer adapts to new behavior pattern automatically, hence minimizing human intervention.

5 Evaluation

5.1 Experimental Setup

Effectiveness and Robustness of our approach were evaluated under both indoor and outdoor surveillance environments. Three different indoor scenarios were set up for experiments. In scenario *A*, a webcam was mounted on the wall of a office as the security camera, monitoring the behavior of people passing by the aisle. Scenario *B* is similar with scenario *A*, but the camera was mounted at the end of the aisle instead. Scenario *C* simulated the arrangement of a real indoor CCTV system, the camera was mounted on the ceiling of the entrance/exit corridor — most of the people would just walk straight and enter the office, a few of them entered the control room on one side of the corridor, which is regarded as an unusual behavior.

In the outdoor scenario *D*, the security camera was assigned to monitor the entrance of a building. The objective of this outdoor experiment is to evaluate the effect of illumination and occlusion on the detection performance. All the experiments described above were conducted on an dual-core 2 GHz platform, with video resolution 640×480 pixels, $N_d = 8$, $N_v = 16$, $S_b = 400$, $S'_b = 600$.

5.2 Analysis

The proposed detection algorithm demonstrates satisfactory results: in both indoor and outdoor case, this approach can successfully detect unusual activities in real-time with high robustness, while making only a few false negatives.

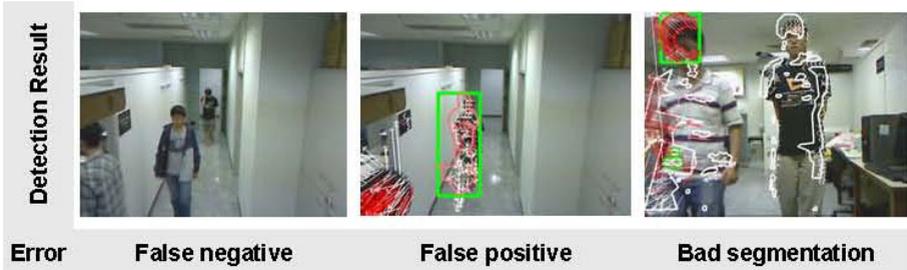


Fig. 8. Examples of incorrect detections

Figure 7(a) illustrates some detection results obtained from the first two indoor environments. In scenario *A*, without providing any labeled samples, the algorithm is able to discover unusual behaviors automatically. However, some unusual behaviors in the scenario *B* cannot be detected completely because of bad background/foreground segmentation as depicted in Figure 8. It is expected that the bad segmentation is due to the slow movement of objects in the scene, misinterpreting some foreground regions as background. Some experimental results in scenario *C* are shown in Figure 7(b). It is shown that the proposed algorithm can also detect unusual behaviors which happen suddenly: on the rightmost column in Figure 7(b), the man on the right hand side was bumped by the man in the middle suddenly, however the algorithm was able to notice the change and made the right response.

Some evaluation results of scenario *D* are recorded in Figure 7(c). Besides detecting unusual movements, abnormal actions can also be detected by the proposed approach. In addition, this approach also shows high robustness against occlusions, unusual behavior can still be detected when the silhouette is greatly distorted by occlusions.

6 Conclusion and Future Works

We presented a novel unsupervised approach to detect unusual behavior for surveillance. This method is specialized for biometric-assisted surveillance as it is able to perform detection in real-time, in order to identify the target immediately by means of face recognition or gait analysis. This approach needs neither a labeled training set, nor a well-defined behavioral definition to detect unusual behavior automatically and incrementally. In future work, we intend to improve the segmentation accuracy by designing a new background model. We also seek to investigate the possibility of introducing *independent component analysis* (ICA) or *supporting vector machine* (SVM) into the current detection algorithm.

Acknowledgement

The work described in this paper was substantially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. 415207).

References

1. Zhou, X., Collins, R.T., Kanade, T., Metes, P.: A master-slave system to acquire biometric imagery of humans at distance. In: IWVS 2003: First ACM SIGMM international workshop on Video surveillance, pp. 113–120. ACM, New York (2003)
2. Nishiyama, M., Yuasa, M., Shibata, T., Wakasugi, T., Kawahara, T., Yamaguchi, O.: Recognizing faces of moving people by hierarchical image-set matching. In: IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR 2007, pp. 1–8 (June 2007)
3. Chan, F., Chen, J., Moon, Y.S.: Fast construction of object correspondence in stereo camera system: an example to human face capturing system. In: IEEE Workshop on Motion and video Computing, 2008. WMVC 2008, pp. 1–6 (January 2008)
4. CNN: Police to give bomb probe details (July 2005), <http://www.cnn.com/2005/WORLD/europe/07/11/london.attacks/index.html>
5. BBC: 7/7 bombers surveillance shown (April 2008), <http://news.bbc.co.uk/1/hi/england/london/7338301.stm>
6. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI 1981, pp. 674–679 (1981)
7. Bouguet, J.Y.: Pyramidal implementation of the lucas kanade feature tracker: Description of the algorithm. Jean-Yves Bouguet (2002)
8. Xiang, T., Gong, S.: Activity based surveillance video content modelling. *Pattern Recogn.* 41(7), 2309–2326 (2008)
9. Xiang, T., Gong, S.: Video behavior profiling for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(5), 893–908 (2008)
10. Zhong, H., Shi, J., Visontai, M.: Detecting unusual activity in video. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, vol. 2, June 2–July 2004, pp. II–819–II–826 (2004)
11. Andrade, E., Blunsden, S., Fisher, R.: Hidden markov models for optical flow analysis in crowds. In: 18th International Conference on Pattern Recognition, 2006. ICPR 2006, vol. 1, pp. 460–463 (2006)
12. Wu, X., Ou, Y., Qian, H., Xu, Y.: A detection system for human abnormal behavior. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005 (IROS 2005), pp. 1204–1208 (August 2005)
13. Li, Y., Xu, C., Liu, J., Tang, X.: Detecting irregularity in videos using kernel estimation and kd trees. In: MULTIMEDIA 2006: Proceedings of the 14th annual ACM international conference on Multimedia, pp. 639–642. ACM, New York (2006)
14. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, p. 2246 (1999)
15. Bobick, A., Davis, J.: The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(3), 257–267 (2001)