Event analysis on TRECVid 2008 London Gatwick dataset

Murtaza Taj, Fahad Daniyal, and Andrea Cavallaro

Queen Mary, University of London
Mile End Road, London E1 4NS (United Kingdom)
{murtaza.taj,fahad.daniyal,andrea.cavallaro}@elec.qmul.ac.uk
http://www.elec.qmul.ac.uk/staffinfo/andrea/

Abstract. In this paper we perform event analysis on a challenging surveillance dataset without any artificial events. We analyze low-level and high-level features such as motion vectors, change detection and pedestrian detections for recognition of events. We performed detection of three events namely person runs, elevator no entry and opposing flow. The event detection is performed on 60 hours of TRECVid 2008 event detection dataset.

1 Introduction

Activity recognition in real surveillance scenarios has gained significant importance in recent years. It can be used for real-time event detection as well as for video summarization and retrieval. Real-time event detection can help to improve the security of public areas by generating alerts and warnings or by assisting security personnel in selecting the camera where something interesting is happening. The summarization and retrieval capabilities can greatly reduce the time required to examine the recorded videos for analyzing events that have already occurred. Similarly, in sports scenarios, summarization can help in generating highlights and video summaries. Considerable amount of work has been done in analyzing activities in simpler datasets (KTH [6], Weizmann [1], Ballet [4]) where the actions are performed in controlled scenarios. The real challenge lies in applying such algorithms [3–5] in scenarios where complexity increases manyfold and the features on which these algorithms rely, such as corner points, optical flow, tracks and shape, may not be readily available due to varying target sizes, occlusion, low video quality and lighting conditions.

In this paper we analyzed events in real surveillance videos where uncontrolled activities were performed. The paper is organized as follows: in Section 2 we discuss some of the features that were extracted to analyze the activities. Section 3 discusses how these features are used and the obtained results. Finally in Section 4 we draw our conclusions and give some future directions.

2 Features

A combination of low-level features and a high-level feature is used for action recognition. The low-level features include motion vectors and foreground segmentation, whereas the high-level feature is the output from a pedestrian detector [9, 10].

The *motion vectors* are computed by applying block matching using different window sizes based on the camera perspective. We use rectangular blocks instead of square block as the target objects, i.e. pedestrians, form upright rectangular bounding boxes. The three different block size used were 2×4, 4×8 and 8×16 with a shift of 1 pixel and a search window of 14×14 pixels.

Video object extraction (foreground segmentation) is performed using a statistical color change detector [2], a model-based algorithm that assumes additive white Gaussian noise introduced by the camera. The noise amplitude is estimated for each color channel separately. Given a reference image (i.e., an image without objects or an image generated by an adaptive background algorithm [8]) the algorithm removes the effect of the camera noise based on the hypothesis that the additive noise affecting each image of the sequence follows a Gaussian distribution with mean zero and standard deviation σ_t. The value of σ_t is computed by analyzing the image difference in areas without moving objects. The foreground is computed by analyzing the image difference $d(i,j) =\mid I^{ref}(i,j) - I^t(i,j) \mid$, where I^{ref} and I^t are the reference and the current image, respectively. The classification between foreground and background pixels is performed based on a dynamic threshold, automatically computed based on noise modeling. This method verifies when $d(i,j) \neq 0$ because of the camera noise as opposed to other factors like moving object or illumination changes. Based on this hypothesis, H_0, the conditional probability density function $f(d(i,j)|H_0)$ is defined as

$$f(d(i,j)|H_0) = \frac{1}{\sqrt{2\pi\sigma_t^2}} e^{-\frac{d^2(i,j)}{2\sigma_t^2}}. \quad (1)$$

The above model is applied on groups of pixels as $f_{\Xi^2}(\Xi^2(i,j)|H_0)$, where $\Xi^2(i,j) = \Sigma_{(k,l)} \in W^n(i,j)d^2(k,l)$ and $W^n(i,j)$ is a square window centered in (i,j) and containing n pixels. After classification, any isolated noise is removed using morphological operators (dilation and erosion).

To *detect people* we use an Adaboost feature classifier based on a set of Haar-wavelet like features [9, 10]. These features are computed on the integral image $\mathcal{I}(x,y)$, defined as $\mathcal{I}(x,y) = \sum_{i=1}^{x}\sum_{j=1}^{y} I(i,j)$, where $I(i,j)$ represents the original pixel intensity. The Haar features are differences between sums of all pixels within sub-windows in the original image. Therefore, in the integral image they are calculated as differences between the top-left and the bottom-right corners of the corresponding sub-windows.

Figure 1 shows the magnitudes of motion vectors. The peaks in the signal indicate activity intervals where there are some objects in the scene. Due to perspective, the magnitude of the motion vectors varies across the scene due to the distance to the objects in view. This magnitude is normalized by dividing

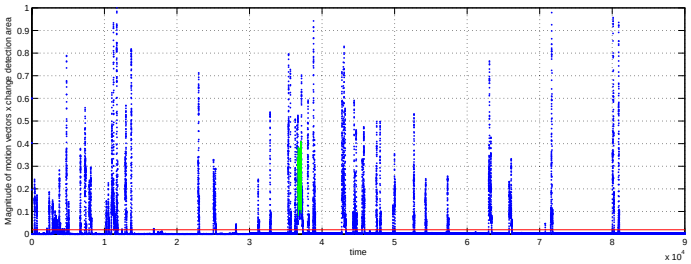

Fig. 1. *Sample motion vector magnitudes in foreground regions. The magnitude below the red line is due to noise only whereas the green patch indicates the interval occupied by the event.*

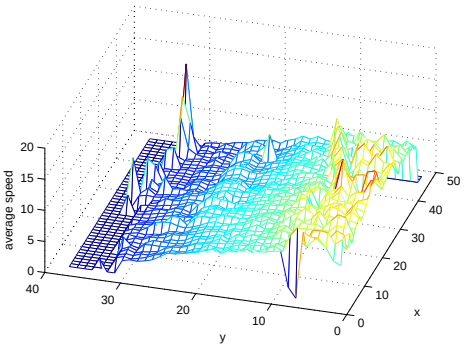

Fig. 2. *Sample normalization factor computed for each* 16×16 *region of the image.*

with the average magnitude, over non-event intervals of each 16×16 block of the scene (Fig. 2). The normalizing factor is further smoothed by applying mean filter.

3 Experimental results and analysis

The event are detected on 60 hours (10 hours of development and 50 hours of evaluation dataset) of the London Gatwick airport dataset. The dataset consist of 5 semi-overlapping cameras.

3.1 Person running

The *person runs* event is detected by analyzing the normalized magnitude of the motion vectors after applying temporal smoothing. Figure 3 shows sample true positive (TP), false positive (FP) and false negative (FN) events. The false positive (FP) (Fig. 3(d-f)) is due to the vehicle moving at a speed higher than the speed of the pedestrians. The people inside the vehicle and the rectangular

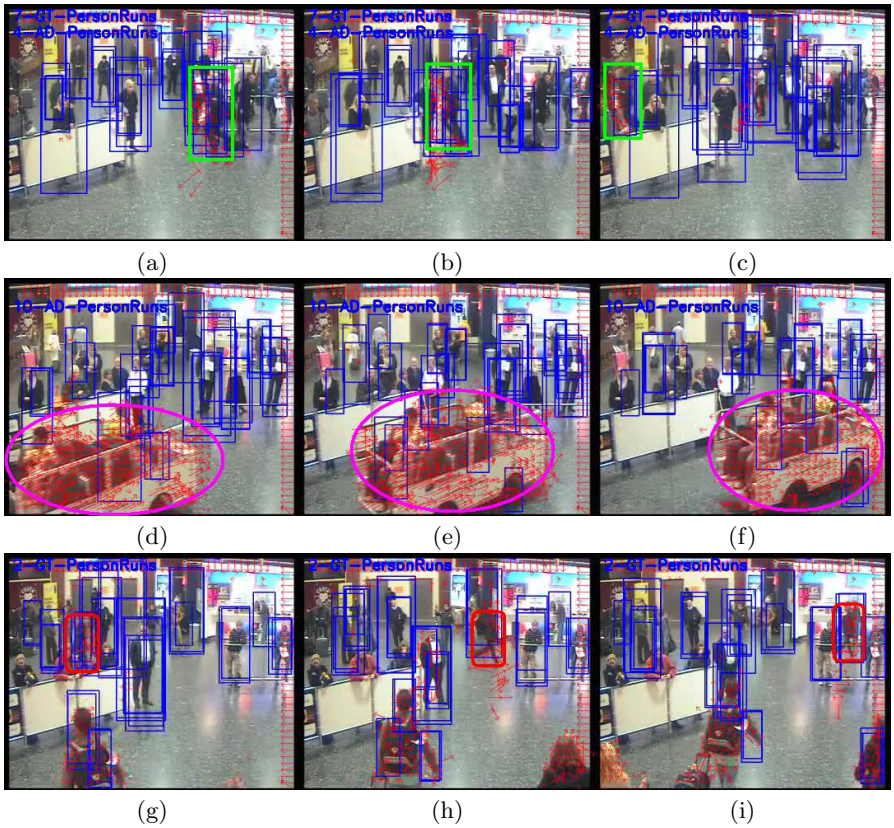

Fig. 3. *Sample* person runs *event detection results. (a-c) True positive; (d-f) false positive; and (g-i) false negative. (For visibility, object associated with true positives are marked with green rectangles, false positives are marked with magenta ellipses and false negatives are marked with red rounded rectangles).*

windows of the vehicle have resulted in several detections by the pedestrian detector, resulting in detection of fast moving motion vectors under a detection window. These pedestrian detections may be correct, but the pedestrians are moving at a higher speed not because they are running but because they are inside the vehicle. The false negative (FN) shown in Fig. 3(g-i) is because the person is running in the far field of the camera where he is mostly occluded and has attracted few motion vectors.

3.2 Elevator no entry

The normalized magnitude of the motion vectors along with the change detection mask are used to detect *elevator no entry* event. Semantic information about locations of elevator doors is assumed to be available as regions of interest. Figure 4 shows sample true positive (TP) and false positive (FP) events, whereas

Fig. 4. *Sample* elevator no entry *event detection results. (a-c) True positive; and (d-f) false positive.*

there are no false negatives (FN) in case of the *elevator no entry* event. In Fig. 4(a-c) the person is standing in front of the right side elevator, whereas the left side elevator is available for use. The person did not used the left elevator for which the doors opened and then closed without anyone entering, hence it is detected as an *elevator no entry* event. The false positive shown in Fig. 4(d-f) is due to the person walking randomly in front of the elevator while talking on a mobile phone. The walking of the person in front of the elevator door is detected as *elevator door activity* while no person is detected to be entering the elevator which resulted in false detection of *elevator no entry* event.

3.3 Opposing flow

The direction of the motion vectors within the detection bounding boxes, inside the region of interest (door region), is used to detect *opposing flow* events. One of the challenges in this event is that the region of interest is in the far field of the camera with high occlusion due to large number of people in a relatively small area. The targets crossing the door from the wrong side are visible in the scene only when they are crossing the doors, hence no information about target motion is available to analyze its possible direction. Motion of a person from the right side of the scene to the left side near the doors can be considered as *opposing flow*. Figure 5(a-c) shows correct detection of the *opposing flow* event despite heavy occlusion. The false detection shown in Fig. 5(d-f) is due to a person going in the opposite direction in front of the doors. The detected bounding box of the person is of incorrect size, because of which the base of the bounding box

Fig. 5. *Sample* opposing flow *event detection results. (a-c) True positive; (d-f) false positive; and (g-i) false negative. (For visibility, objects associated with true positives are marked with green rectangles, false positives are marked with magenta ellipses and false negatives are marked with red rounded rectangles).*

is inside the door regions and hence is detected to be an *opposing flow* event. Since in truth the person didn't, actually cross the door, it is a false detection. The event shown in Fig 5(g-i) is very similar to the correct detection shown in Fig. 5(a-c). The difference here is that the color of the clothes is similar to the background color and therefore very few motion vectors are detected. The increased crowd in this case also resulted in missed detection by the pedestrian detector. The failure of both the motion vectors and the pedestrian detector contribute to false negatives in this case.

3.4 Evaluation

The detection scores shown in Table 1 were computed using the TRECVid 2008 evaluation metrics [7]. There is no score for *elevator no entry* as there are no occurrences of this event in the dataset. The scores indicate a significant increase

Table 1. *Detection scores for* person runs*,* elevator no entry *and* opposing flow *events on development and evaluation datasets of London Gatwick airport. Note: There is no score for* elevator no entry *event on evaluation data as there are no occurrences of this event in the evaluation dataset*

	Person runs	**Elevator no entry**	**Opposing flow**
Development data	0.1665	0.584	0.2614
Evaluation data	0.8012	NA	0.2014

in *person runs* detections, but a slight decrease in *opposing flow* detection performance, from development data to evaluation data. This is because the metric penalizes missed detections 10 times more than false detections. The runs on the evaluation data are therefore tuned to generate a smaller number of missed detections but at a cost of generating more false detections.

4 Conclusions

In this paper we have analyzed events on a real surveillance dataset from London Gatwick airport. The events were analyzed using both low-level and high-level features. The combination of both low and high level feature is required to detect events. In future we plan to train a classifier on these features, to improve detection scores. We plan to analyze cuboids which are temporal windows over the scene that can incorporate both spatial as well as temporal information. The cuboid features then can be trained using a classifier such as SVM or Adaboost.

References

1. M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Proc. of IEEE Int. Conf. on Computer Vision*, volume 2, pages 1395–1402 Vol. 2, October 2005.
2. A. Cavallaro and T. Ebrahimi. Interaction between high-level and low-level image analysis for semantic video object extraction. *EURASIP Journal on Applied Signal Processing*, 6:786–797, June 2004.
3. A.A. Efros, A.C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proc. of IEEE Int. Conf. on Computer Vision*, pages 726–733 vol.2, October 2003.
4. A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.
5. J.C. Niebles and Li Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.
6. C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *Proc. of IEEE Int. Conf. on Pattern Recognition*, volume 3, pages 32–36 Vol.3, August 2004.

7. A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.
8. C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Machine Intell.*, 22:747–757, August 2000.
9. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 511–518, Kauai, Hawaii, December 2001.
10. P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proc. of Int. Conf. on Computer Vision Systems*, volume 2, pages 734–741, October 2003.