SEGMENTATION OF MOVING PEDESTRIANS WITHIN THE COMPRESSED DOMAIN

Miguel Coimbra, Mike Davies

Miguel Coimbra and Mike Davies
Queen Mary College (University of London)
Department of Electronic Engineering
miguel.coimbra@elec.qmul.ac.uk; mike.davies@elec.qmul.ac.uk

ABSTRACT

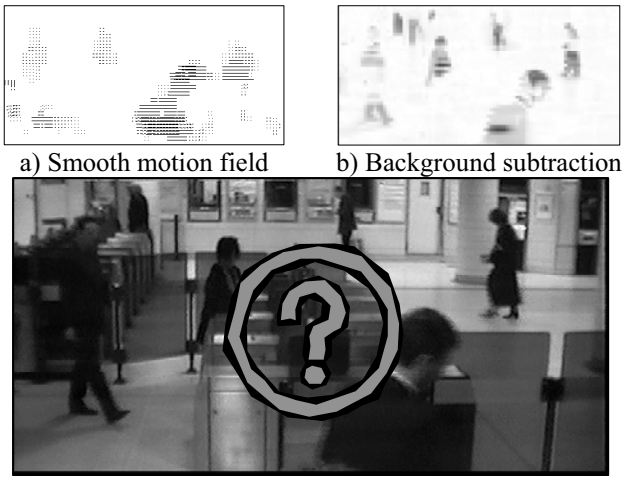
Video encoding standards, namely MPEG-2, store large amounts of information obtained for compression purposes that can be accessed with minimal decoding. This paper shows that, with proper filtering of motion vectors and DCT coefficients, accurate segmentation results can be achieved by combining both reliable motion estimation and background subtraction. We further present a fine segmentation step that exploits specific blob characteristics to reduce segmentation noise and solve some occlusion problems. Examples using real videos from underground station CCTV cameras show that compressed domain information can be the key for successful surveillance applications where very fast algorithms with high accuracy are required.

1. INTRODUCTION

Segmentation is one of the most complex challenges of image processing. The literature is full of different techniques that can be characterized by their accuracy, speed, scope, etc. A research area such as surveillance has specific requirements from segmentation algorithms. Speed is crucial for real-time event detection but these fast results need enough accuracy so that events are actually recognized. Some assumptions simplify this task such as static cameras and generally controlled environments. A good example of a commercial product using image domain background subtraction is Fuentes and Velastin's work [1].

With the growth in popularity of digital video, compressed domain techniques have been becoming more popular since working on these reduced data spaces allows for very fast processing. Segmentation has been attempted before either using only motion information [2] or by combining it with background subtraction [3]. Both achieve reasonable results on simple situations but to extend them to more complex scenes requires the use of more reliable motion estimations such as the one presented by the authors previously [4].

An initial improved segmentation is here presented, followed by a fine segmentation step that obtains better results by both reducing noise and handling some occlusion situations. The proposed methods will be presented in section 2 and results in section 3. In section 4 these results will be discussed and conclusions drawn.



a) Smooth motion field b) Background subtraction

c) Original image

Figure 1 – Available compressed domain information

2. METHODS

Previous work by the authors has shown how reliable motion estimation can be obtained from the MPEG-2 compressed domain [4]. A set of normalization rules, followed by a 3x3 spatial median filter obtains a *smooth motion field* (*SMF*) (Fig. 1.a) where one motion vector is provided for each 16x16 block, independently of MPEG picture type. This motion estimation is then refined by an optical flow *confidence map (CM)*, obtained from the AC[1] and AC[8] coefficients of the DCT transform.

Shen and Delp [5] have shown how low resolution images can be obtained for P and B-frames of an MPEG sequence from the DC coefficients of the DCT transform. Compressed domain background subtraction (Fig 1.b) can

then be used on these DC images and combined with the motion estimation results to segment moving pedestrians. The background subtraction segmentation (*BBS*) mask, is obtained by equation (1) where *DC* is the DC image and *BKG* is the previously learnt background image.

$$BSS(x,y) = \begin{cases} 1 \Leftarrow |DC(x,y) - BKG(x,y)| \geq k_b \\ 0, otherwise \end{cases} \quad (1)$$

Motion segmentation *MS* is achieved in a similar manner as equation (2) shows, by using a threshold k_m on the vector magnitude from the *smooth motion field (SMF)*.

$$MS(x,y) = \begin{cases} 1 \Leftarrow SMF(x,y) \geq k_m \\ 0, otherwise \end{cases} \quad (2)$$

Binary multiplication of both these results obtains our initial segmentation *S* in equation (3).

$$S(x,y) = BSS(x,y).MS(x,y) \quad (3)$$

Connected component analysis on matrix *S* provides us with a set of *blobs* B_i that, given the high quality of the information, show us the areas with moving objects. A fine segmentation of these blobs is possible by analysis of several of its characteristics. Fig. 2 shows a diagram of how this is obtained.

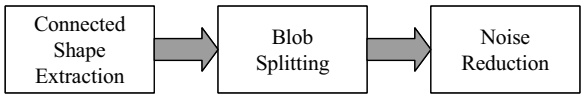


Figure 2 – Details of the fine segmentation step

After blob segmentation, the following characteristics were analyzed: blob size, blob height, blob width, blob height/width ratio, bounding box size, motion direction, confidence size (number of high confidence motion vectors), confidence size / blob size ratio, vertical histogram and confidence vertical histogram.

A set of test sequences were segmented using the previously described techniques and blobs were manually classified as pedestrians, group of pedestrians, partial pedestrians and noise. All the characteristics were then analyzed and its ability to detect these situations measured. This study concluded that the most important characteristics for removing false detections, that is, the ones that more accurately distinguished between noise and pedestrian blobs, were *blob size* and *confidence size*. The best for occlusion handling (distinguishing pedestrians from groups of pedestrians) were *motion direction* and *vertical histogram*. Following this, the fine segmentation stage can be summarized in the diagram of Fig. 3. Motion direction and vertical histogram are used for blob splitting. Noise reduction is obtained by ignoring blobs with very low size and confidence.

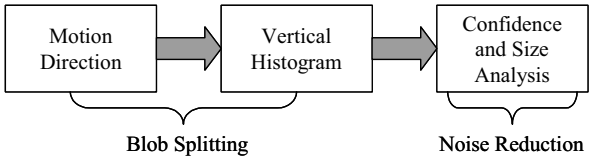


Figure 3 – Steps of the fine segmentation stage

3. RESULTS

This system has been implemented on a Pentium 1GHz. All mpeg sequences were encoded using an MVCast Mpeg-2 A/V Encoder. Image size is 704x480 with 4:2:0 chroma format and IBBPBBPBBPBB GOP format. A total of 36 video sequences with an average duration of 4 seconds were tested. These were obtained from 9 different surveillance cameras that vary in camera angle, pedestrian density and illumination conditions so results could generalize to a vast range of situations.

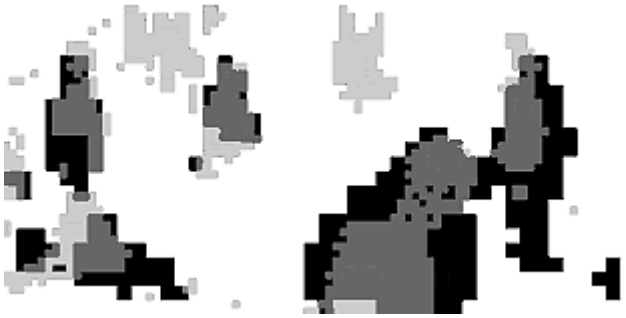


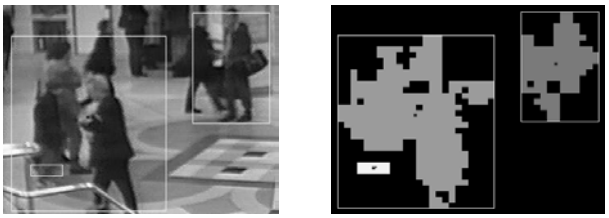
Figure 4 – Initial segmentation results

Fig. 4 shows a typical example of segmentation using *BSS* (light gray), *MS* (black), and both *BSS* and *MS* (dark gray). The following sub-sections will focus on individual blob properties that allow us to improve these initial segmentation results. Numerical evaluation consists in measuring the number of groups, pedestrians, partial detections and noise for each stage of the fine segmentation. Occlusion is reduced when the number of detected pedestrians increases and the number of group detections decreases. Noise reduction is measured as a decrease in the number of noise detections. Partial detections are not considered in this study.

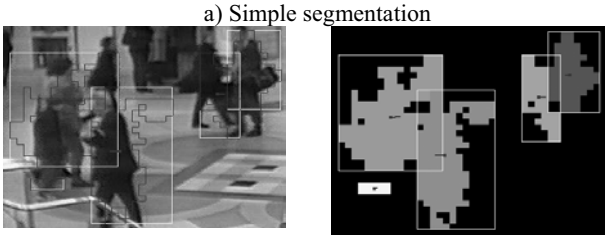
3.1. Horizontal motion direction

The *smooth motion field* obtained uses a translational motion model, defined by two parameters: horizontal and vertical displacement. Such method handles object translation effectively but has problems when facing rotation and zoom. In our application scenario this means that results are very accurate in cameras where pedestrians move horizontally but poor results are obtained in cameras placed in low ceilings with pedestrians moving towards or away from the camera since displacements

resemble zoom operations. We therefore concentrate on scenes dominated by translational movement here.



a) Simple segmentation



b) Using motion direction

Figure 5 – Blob splitting using horizontal motion direction

Instead of producing one binary motion segmentation matrix *MS*, two are generated MS_1 and MS_2: one using only vectors pointing left (MS_1) and one with vectors pointing right (MS_2). Using equation 3 two segmentation matrices S_1 and S_2 are thus obtained and connected component analysis is applied individually to each one to obtain the set of blobs B_i.

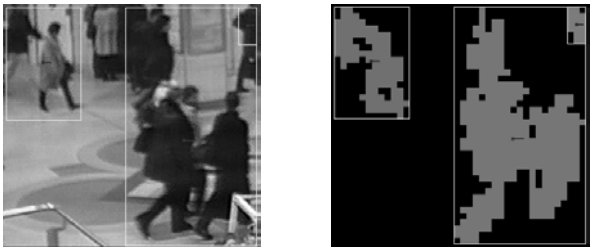
This computationally simple method solves a large number of occlusion situations. Fig. 5.a shows the segmentation using no horizontal motion and it can be seen that Fig 5.b, with this information, handles some of these occlusion problems. Numerical analysis (Table 1) shows this improvement in the increase in pedestrian detections from 34 to 41.

3.2. Vertical histogram

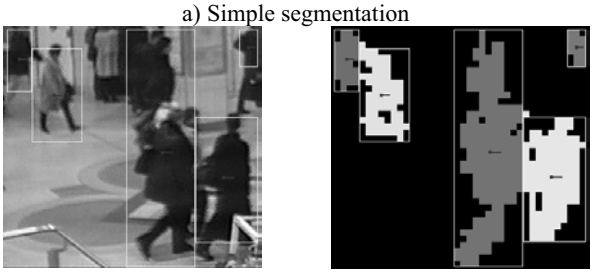
An analysis of the blob characteristics showed that typically, the vertical histogram of a pedestrian's shape is bell-shaped with a single vertical peak and no local minima. Group blobs do not have such regular patterns and usually have histogram shapes with at least one local minima. Using equation 4, groups can be separated into two new blobs when this local minima *m* is found in the vertical histogram *h(x)*.

$$\forall_{x\in[m-2,m+2]} h(m) \le h(x) \tag{4}$$

The proposed method searches for this local minima and splits the blob. Segmentation results of Table 1 shows that the number of detected pedestrians increases by 4 using this method. Fig. 6 shows an example of a situation where using vertical histogram solves some occlusion problems. Fig. 7 shows the corresponding vertical histograms and the detected minima. The darker part of the bars shows the amount of high-confidence vectors present in each column.



a) Simple segmentation

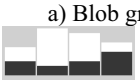


b) Using motion direction

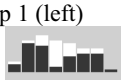
Figure 6 – Blob splitting using vertical histogram

a) Blob group 1 (left)

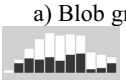
a) Blob group 2 (right)



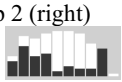
c) Left pedestrian



d) Right pedestrian



e) Left pedestrian



f) Right pedestrian

Figure 7 – Vertical histograms of previous example

3.3. Confidence and size analysis

As previous studies show, motion vector confidence *MVC* can be used to reduce illumination noise [4]. A blob B_i with near zero confidence is most probably noise caused by shadows and reflections thus it should be removed. Blob confidence is defined in equation (5).

$$BlobConf_i = \sum_{x,y} CM(x,y).B_i(x,y) \tag{5}$$

Another characteristic used is blob size. A study on blob characteristics showed that, in all sequences, the minimum pedestrian blob size was 11. This means that we can safely reject all detections below this value therefore reducing segmentation noise significantly. Blob size is defined in equation (6).

$$BlobSize_i = \sum_{x,y} B_i(x,y) \tag{6}$$

Rejecting blobs with one or zero confidence as well as blobs with size lower than the minimum size obtained for all 9 cameras achieves significant noise reduction. In the example sequence, summarized in Table 1, noise blobs decrease from 171 to 2. It is important to stress the superior performance of *BlobConf* when compared to *BlobSize*. If used individually, *BlobSize* eliminates 97.0% of noise blobs but also incorrectly eliminates 20 partial blobs (10.7% of the total eliminated blobs). *BlobConf* has a slightly superior performance in noise reduction achieving 98.2%. Unlike blob size, it only rejects 12 partial blobs (6.6% of the total eliminated blobs) thus

obtaining higher accuracy while generating less false alarms. The total segmentation noise reduction of this processing stage is 98.8%.

3.4. Final segmentation results

Using all these characteristics in the proposed fine segmentation step reduces a large quantity of false detections and handles several occlusion scenarios. Results for the example sequence are summarized in Table 1 and a sample image is shown in Fig. 8. All the other sequences exhibited similar behaviors. A final example, using a crowded scene shows the importance of this analysis. Without a fine segmentation step only 3 large blobs are obtained as Fig. 9.a shows but using the proposed techniques a large number of individual pedestrians can be detected (see Fig.9.b and Fig. 9.c).

	Noise	Partial	Pedestrians	Group
Initial Segmentation	154	43	34	14
Motion Direction	167	48	41	13
Vertical Histogram	171	46	45	12
Full Segmentation	2	28	45	12

Table 1 – Segmentation results

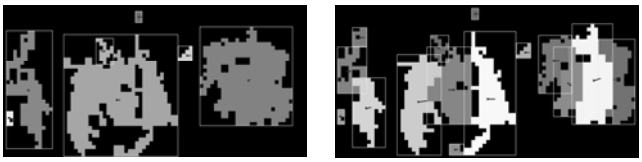
4. CONCLUSIONS

A pedestrian segmentation method using motion vectors and DCT coefficients has been proposed, with emphasis on computational simplicity. The system consists of three main components: motion estimation; background subtraction; and a segmentation refinement stage. Since compressed domain information is used, this segmentation is obtained with minimal computational costs in comparison to traditional image domain techniques.
Furthermore, all the results of the proposed system are obtained using information from a single image. This means that several occlusion problems are solved without a tracking stage. Further significant improvements are anticipated if tracking techniques (Kalman filtering, maximum overlap, etc…) were to be included.

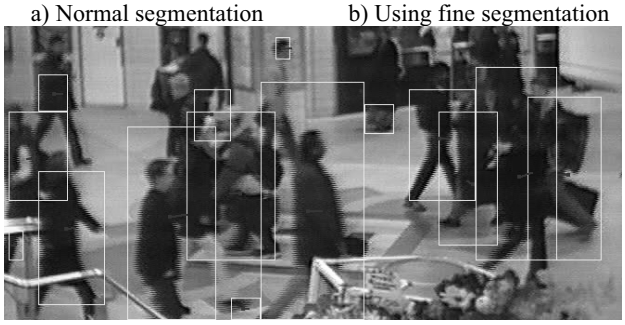
ACKNOWLEDGMENTS

This research has been supported by Fundação para a Ciência e Tecnologia. The authors would like to thank Sergio Velastin from Kingston University and all the members of the Queen Mary College DSP group for the useful discussions and comments.

Figure 8 – Fine segmentation results



a) Normal segmentation b) Using fine segmentation



c) Real image

Figure 9 – Segmentation in a crowded situation

REFERENCES

[1] L. Fuentes, and S. Velastin, "From Tracking to Advanced Surveillance", in *Proc. IEEE ICIP*, Spain, 2003.

[2] M. Coimbra, M. Davies, and S. Velastin, "Pedestrian Detection using MPEG-2 Motion Vectors", in *Proc. of IEEE WIAMIS*, London, 2003, pp. 164-169.

[3] X. Yu, L. Duan, and Q. Tian, "Robust Moving Object Segmentation in the MPEG Compressed Domain", in *Proc. IEEE ICIP*, Spain, 2003.

[4] M. Coimbra, and M. Davies, "A New Pedestrian Detection System using MPEG-2 Compressed Domain Information", in *Proc. IASTED VIIP*, Spain, 2003, pp. 598-602.

[5] K. Shen, and E.J. Delp, "A Fast Algorithm for Video Parsing using MPEG Compressed Sequences", in *Proc. of IEEE ICIP*, 1995, pp.252-255.