MOTION SENSOR FOR VIDEO SURVEILLANCE IN THE MPEG DOMAIN

Rui Marcelino[1,2], Vitor Silva[2] and Sérgio Faria[2,3]

[1]School of Technology, University of Algarve, Faro, Portugal
[2]Institute of Telecommunications, DEEC, University of Coimbra, Coimbra, Portugal
[3] School of Technology and Management, Polytechnic Institute of Leiria, Leiria, Portugal

ABSTRACT

The use of surveillance systems are continuously increasing and the amount of processed and stored data has risen exponentially. The main problem is related to the huge amount of data to process, either by a human operator or by a computer. We have addressed this problem, developing a method for automatic inspection of compressed video bit streams, that is able to detect motion in the scene and camera motion. These features can be detected very fast and with low computational complexity, as only few parameters of the compressed data are processed.
Also this method takes advantage of the characteristics of some surveillance videos, namely those placed in parking areas or warehouses, outside of the working time period, when, most of the time, there are no significant motion and the difference between consecutive video frames is very low. The proposed algorithm is able to separate real motion in the scene from environmental interference or camera noise. For motion detection, the algorithm mainly relies on the type and amount of coded macroblocks, occurring within inter coded frames of a MPEG bit stream.

1. INTRODUCTION

Video sensors with digital compressed domain interfaces are a key tool in modern surveillance system architectures [1]. They allow remote sensing over data networks and the processed data is easily recorded and integrated in computer applications, for helping human operators in supervision tasks.

A typical surveillance scenario may include a video camera, or a set of cameras, pointing to a quite zone, for long periods of time. Thus, most of the time, the received visual information is useless. However, when an important image change occurs, like a human intrusion, the operator should be automatically warned and the recording system must be triggered. When these cameras are installed in remote places, video compression is required, prior to the transmission. At the receiver, the compressed signal (bit stream) can be either decompressed and visualized by an operator or stored. Usually it has to be decompressed until the pixel domain, either for human visualization or to be processed for feature extraction. Otherwise, most of the feature extraction can be performed using parameters included in the compressed data, without completely decompressing the video bit stream. This task can be performed by an efficient and intelligent method using a motion detector.

Several methods and approaches for motion detection have been proposed in the literature [2]. Motion detection can be performed in pixel domain, or uncompressed domain, using features extracted from the digital spatial-temporal video representation, with very computational demanding techniques [2], [3].

Since we assume the digital video signal is compressed using MPEG 1, 2 or 4 standards [4], [5], the motion detection can be done using the information embedded in the bit stream [2], [6], [7], [8], [9]. By partially decoding (parsing) the coded bit stream, one can find quickly and easily the necessary information to build a motion detector [6]. Such techniques are designated by compressed domain solutions.

The main purpose of this work is to use, as much as possible, the embedded information, taking advantage of the huge amount of analysis work performed by the MPEG video encoder.

In the following sections we will describe an efficient and low complexity scene change detector algorithm, which is able to detect significant visual events from a partially decoded MPEG bit stream. Also, we can use the proposed method for a fast search in a posterior analysis of the recorded video. In section 2 we introduce MPEG standard and in section 3 the proposed algorithm is described. Some results are shown in section 4 and conclusions are presented in section 5.

2. MPEG OVERVIEW

MPEG encoders use a hierarchical and hybrid algorithm to compress video, by classifying and processing each frame as intra coded (I frame) or motion compensated inter coded (P and B) [4], [5]. The intra frame pictures are encoded only using pixels within a frame, exploring the spatial redundancy. The P frames are encoded using motion compensated prediction from a past I/P frame, in order to remove the temporal redundancy. The B frames are encoded using motion compensation prediction from both past and/or future encoded I/P frames. Video frames are organized in regular structures called group of pictures (GOP). A usual GOP structure is: [IBBPBBPBBPBBI].

Each frame is divided into blocks of 16x16 pixels, called macroblocks (MB). Furthermore, each macroblock is divided into six 8x8 pixel block's, whose are DCT decomposed and entropy coded. A macroblock contains information about the type of temporal prediction used (or not) for motion compensation. It can be intra coded, forward referenced, backward referenced, interpolated or direct.

While MBs inside an I frame are intra coded, each MB in a P frame is either forward predicted, intra coded or skipped. Similarly, each MB in a B frame is either forward predicted, backward predicted, bidirectionally predicted, intra coded or skipped.

3. COMPRESSED DOMAIN MOTION DETECTION

Considering a video sequence obtained by a digital camera, from a scene without motion, we may assume the following model,

$$I_t[m,n] = I_{t-1}[m,n] + n_t[m,n]\,, \qquad (1)$$

where $I_t[m,n]$ is the input image at instant t and $n_t[m,n]$ is the camera noise, given by a zero mean stationary Gaussian random process [3].

Ideally, a video sequence encoded by MPEG, according to model (1), would be formed by GOPs with I frames only (all inter frames are skipped or not coded). Obviously, in a real environment, there is always a residual number of coded MB, mainly, due to the camera noise, motion and some intrinsic characteristics of the encoder.

3.1 Proposed algorithm

In the context of the motion model described above, the number of coded MBs found in inter frames (P and B) of a GOP is an efficient indicator of scene changing, if we assume a static view model [3]. In this case, coded MB means intra coded, forward predicted and backward predicted. The bidirectionally predicted MB (interpolated and direct) are not counted, since they represent a simultaneous high correlation to near past and future reference frames and, thus, low significant change.

If the maximum number of coded MBs within a given GOP, $c[n]$, exceeds an appropriated threshold [10], then a motion detection alarm will be issued.

Usually, a typical GOP duration (0.3, 0.5 seconds) is less than a significant visual event. Therefore, we may assume that MPEG intra frames are not relevant, in opposition to [6], which is a spatial method and, consequently, the new approach is justified.

The proposed method is similar to temporal shot detection algorithms for studio applications [11], [12]. However, in our framework the frame position detection does not require to be very accurate.

After this step, it may happen that some motion detections are false, due to camera switching (scene cuts) or camera motions. These false motion detection events have to be eliminated, in order to increase the algorithm efficiency. Thus, a scene cut detection method is used [9], [10], [11] and the camera motion is detected [12], as explained in section 3.2.

3.2 Camera motion estimation

Camera movements like: pan, tilt, zoom, shaking and vibration, caused by wind in outdoor environments are sources of false positives motion detections. In order to remove such false positives, we have incorporated in our method a camera motion estimation module.

The optical flow field is approximated using motion vectors and the MB type information extracted from the MPEG streams [13]. To increase the field coherence, we have developed a new approach for true motion vector reconstruction, using the available information.

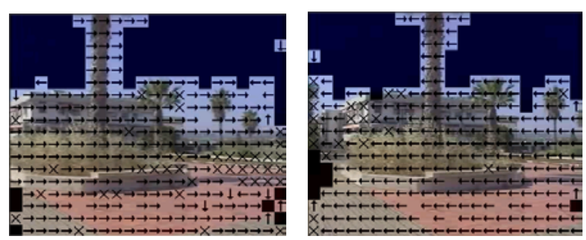

Figure 1. Motion vector's field for a translational camera movement (static background MBs are dark colored).

The motion camera, mentioned above, is well characterized in a frame analysis, by a large number of forward predicted MB and near homogeneous vector fields, as can be seen in Figure 1. This CIF "Palm" sequence represents a translational camera movement to the right (left image) and a similar camera movement to the left (right image), focusing a palm tree. This type of motion generates a large amount of motion vectors, with the same direction of the motion in the scene, and the opposite of the camera motion. When the image texture is not homogeneous, a large number of MBs are encoded with motion vectors, whose intensity depends on the camera motion. Otherwise, when there are homogeneous texture the number of motion compensated (MC) coded MB is reduced, like the sky in the "Palm" sequence, where static background MBs are dark colored.

3.3 Scene classification

The scene classification is an important issue, as it is directly related to the motion detection. This task must be performed by an operator, regarding the surveillance camera system, namely, distance from the scene, zoom lens and target object's size. Thus, we have divided the surveillance scenes in three classes: A, B and C. These classes have a direct correspondence to the number of expected blocks or macroblocks with non-zero motion vectors, within a VOP. The chosen threshold directly determines the detection performance. Due to such direct mutual dependence, the detection performance is highly sensitive to specified parameter values. Beside the threshold sensitivity, the problem of specifying such a precise value remains and, consequently, the scope of the validity of such an accurate threshold is highly questionable. Clearly, manual threshold specification cannot be avoided in practical implementations. Thus, there must have an installation and set-up phase where the sensitivity of the motion sensor must be adjusted. However, the influence of these parameters on the detection performance can be diminished and the detection can be made more robust if we use lower threshold levels. In fact, it is preferable a false alarm rather than a missed alarm.

Figure 2. Test sequences: Hall, Room125, Door125 and Park, in clockwise order.

4. EXPERIMENTAL RESULTS

In this section, we evaluate our motion detector for video surveillance systems. We have performed a series of experiments on videos obtained from surveillance systems installed in the University Campus, which have been encoded in MPEG-4 format.

The length of these videos is between 241 (Pupils), 846 (Park) frames long, and each frame is of size 352x288 pixels. The original sequences, illustrated in Figure 2 as Hall, Room125, Park and Door125 were carefully recorded in order to include many effects, covering the largest number of different situations. The experimental results demonstrate the effectiveness of the proposed motion detection algorithm.

Figure 3. Encoded MBs referenced as black squares.

Figure 3 gives an example of motion detection results in the scenes of various sequences. Note that no filter is used and some MB has been coded due to noise and shadow.

The performance is given in terms of *Precision* and *Recall* parameters [14] given by,

$$Precision = \frac{N_C}{N_C + N_E} \tag{2}$$

and

$$Recall = \frac{N_C}{N_C + N_M}, \tag{3}$$

where N_C means the number of correct motion detections, N_E means the number of incorrect motion detections and N_M means the number of missed motion detections.

In our experiments, the recall and precision values, presented in Table 1, illustrate Precision values very close to 100% for most sequences and a Recall values very close to 90% in most sequences.

Sequence	Length	Precision (%)	Recall (%)
Hall	308	100	87
Room125	679	100	87
Park	846	79	65
Door125	392	97	92

Table 1. Results of Precision and Recall for the tested sequences.

Although the rates of true and missed detections are not precisely the same for all sequences, there are no outliers in the performance. We can say that the performance of our detector remains relatively consistent over all sequences. Those values of Park sequence are related to the appearance in scene of moving objects of distinct classes, which are cars in natural plane of the scene and other cars parking at a long distance of the camera, almost indistinguishable points in the scene.

5. CONCLUSIONS

In this paper we describe an efficient and low complexity motion detection algorithm for surveillance systems, mainly for remote applications where long video sequences are MPEG encoded and recorded. The proposed method parses the compressed data, being able to detect only true motion in a scene. Various techniques have been implemented to detect: scene cuts, zoom, camera translation and camera oscillation. These methods strongly reduce the incorrect motion detection rate. Furthermore, we have presented the used techniques, emphasizing the motion detector, which is based on the type and amount of coded macroblocks, that occur within inter coded frames of a MPEG bit stream.

6. REFERENCES

[1] C. S. Regazzoni, V. Ramesh and G. L. Foresti (Eds.), "Special Issue on Video Communications, Processing and Understanding for Third Generation Surveillance Systems", *Proceedings of IEEE*, vol. 89, nº. 10, October 2001.

[2] I. Koprinska and S. Carrato, "Temporal Video Segmentation: A Survey", *Signal Processing: Image Communication*, vol. 16, pp. 477-500, 2001.

[3] A. Albiol, C. Sandoval, A. Albiol, V. Naranjo and J. M. Mossi, "Robust Motion Detector for Video Surveillance Applications", in *Proc. IEEE ICIP 2003*, September 2003.

[4] John Watkinson*, The MPEG Handbook, MPEG-1, MPEG-2, MPEG-4*, Focal Press, 2001.

[5] F. Pereira and T. Ebrahimi (Eds.), *The MPEG-4 Book*, Prentice Hall PTR, 2002.

[6] J. Pons, J. P. Nebot, A. Albiol and J. Molina, "Fast Motion Detection in Compressed Domain for Video Surveillance", *Electronics Letters*, vol. 38, nº. 9, pp. 409-411, April 2002.

[7] B. Yeo and B. Liu, "Rapid Scene Analysis on Compressed Video", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 5, nº. 6, pp. 533-544, December 1995.

[8] S. Pei and Y. Chou, "Efficient MPEG Compressed Video Analysis Using Macroblock Type Information", *IEEE Trans. on Multimedia*, vol. 1, nº. 4, pp. 321-333, December 1999.

[9] S. Lee, Y. Kim and S. Choi, "Fast Scene Detection using Direct Feature Extraction from MPEG Compressed Videos", *IEEE Trans. on Multimedia*, vol. 2, nº. 4, pp. 240-254, December 2000.

[10] A. Hanjalic, "Shot Boundary Detection: Unraveled and Resolved?", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 12, nº. 2, pp. 90-105, February 2002.

[11] J. Calic and E. Izquierdo, "Temporal Segmentation of MPEG Video Streams", *EURASIP Journal on Applied Signal Processing*, nº. 6, pp. 561-565, 2002.

[12] Y. Haoran, D. Rajan and C. L. Tien, "A Unified Approach to Detection of Shot Boundaries and Subshots in Compressed Video", in *Proc. IEEE ICIP 2003*, September 2003.

[13] R. Wang and T. Huang, "Fast Camera Motion Analysis in MPEG Domain", in *Proc. IEEE ICIP 1999*, vol. 3, pp. 691-694, October 1999.

[14] U. Gargi, R. Kasturi and S. Strayer, "Performance Characterization of Video Shot Change Detection Methods", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 10, nº. 1, pp. 1-13, February 2000.