# Real-Time video-surveillance by an Active Camera

G.L. Foresti and C. Micheloni
*Department of Mathematics and Computer Science (DIMI)*
University of Udine, Via delle Scienze, 206, 33100 Udine, ITALY

**Abstract** In this paper, we propose a real-time video-surveillance system for image sequences acquired by a moving camera. The system is able to compensate the background motion and to detect mobile objects in the scene. Background compensation is obtained by assuming a simple translation of the whole background from the previous to the actual frame. Dominant translation is computed on the basis of the tracker proposed by Shi-Tomasi and Tomasi-Kanade. Features to be tracked are selected according to a new intrinsic optimality criterion. Badly tracked features are rejected on the basis of a statistical test. The current frame and the related background, after compensation, are processed by a change detection method in order to obtain a binary image of moving points.Results are presented in the contest of a visual-based system for outdoor environments.

## 1 Introduction

Detection and tracking of moving objects are important tasks for computer vision, particularly for visual-based surveillance systems [7, 2]. Video surveillance application, most times, imply to pay attention to a wide area, so omnidirectional cameras [4] or mobile cameras [3] are generally used.

In the proposed system, a mobile camera with tunable parameters (i.e., a camera which can change the viewpoint, for example to keep a target in the center of the image, or modify intrinsic parameters like focus or zoom) has been considered. The detection of mobile objects is the main objective of the proposed system. Several works demonstrated that motion detection is a difficult task if image sequences are obtained by a moving camera [1, 5, 8].

To this end, we introduce a new real-time motion detection technique, based on the well known Shi-Tomasi [9] and Tomasi-Kanade [10] tracker, for application to visual-based surveillance systems.

The proposed method is focused particularly on the determination of the best displacement vector, and it differs from the techniques adopted till now. In [11], Tommasini *et al.* used feature rejection rule to eliminate bad tracked features in the median computation of single features displacements. In this case, the displacement calculation is a consequence of the feature rejecting. In [1], the heuristic adopted consists in the determination of three features whose affine transformation parameters are optimal. This involve to compute the parameters for an uncertain number of times, that is in contrast with the real-time constraint.

Our idea is to find , at the first iteration, the set of the features that involve the best displacement estimation. After this, we reject all the features that does not belong to the set just determined. The advantage is the possibility to work in real-time and a bigger accuracy in the displacement estimation can be obtained.

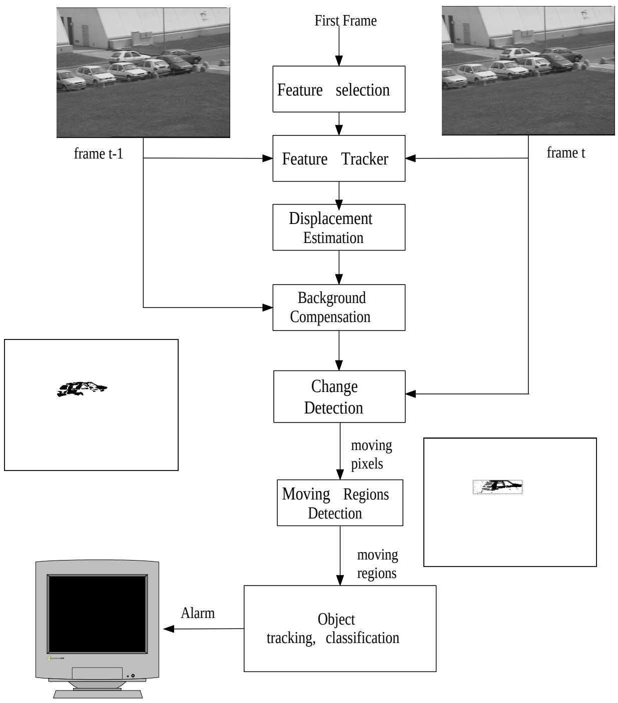



Figure 1: General architecture of the surveillance system

## 2 System Description

As shown in Figure 1, the proposed system is based on a frame by frame motion detection technique. An initialization occurs when the system starts. From the first frame a feature selection, using the method exposed in [9], is necessary in order to built the first set of trackable features.

The tracker developed by Shi-Tomasi [9] and Tomasi-Kanade [10] is applied on the feature set and the feature positions in the current frame are obtained as result. Current positions are compared with the old ones to estimate the displacement among the frames. Results of the displacement estimation are used for two different objectives: background compensation and feature rejection.

The background compensation translates the current frame by the estimated displacement vector **d**. The resulting image has the property that every static pixel is in the same position in the two frames. Only pixels belonging to moving objects are in different positions. Difference images are thresholded to obtain the blob image that approximately, in presence of an high frame-rate, contains the edges of the moving objects (called *moving edeges* [8]). The blob generated is composed by a superimposition of the blob of the moving object in the old and new frame.

The last step of the system consists in the segmentation of the regions belonging to the moving objects.

Moving regions can be used to act a video surveillance purpose. Particularly an object can be detected and tracked maintaining it at the center of the acquired image.

The focus of the paper is on the description of displacement estimation, necessary to build a robust feature tracker and to compensate the background.

## 3 Robust feature tracking

The purpose of our tracker was to operate, in real-time, in most of the real environments situations like indoor and outdoor scenes and in every kind of luminance (acceptable for a CCD camera). These constraints have required to introduce some improvements with respect to existing techniques [1, 11].

Once the set of feature is built, the feature tracking algorithm is used on it. As a result we have correspondences among features in the current frame and in the previous frame that allows to compute their displacement. The problem is that not all the features are tracked well. It is necessary distinguish those tracked well from the others. The idea is to label a feature as tracked well if its displacement corresponds to the image displacement.

The strategy followed to determine the displacement is to define a reliability factor for each displacement present in the feature set. For each feature, a residual computation is normalized in order to limit the effects of intensity changes between frames, by subtracting the average grey level for each window:

$$E = \sum_{W} [(J(\mathbf{x} + \mathbf{d}) - \overline{J}) - (I(\mathbf{x}) - \overline{I})]^2 \tag{1}$$

where $J(\cdot) = I(\cdot, t + \tau)$, $I(\cdot) = I(\cdot, t)$, $\overline{J}$ and $\overline{I}$ are the average grey levels in the two considered regions. The reliability factor is then calculated by adding all residual errors of the feature having the same displacement and weighting the result by dividing it by the number of the interesting features:

$$RF(D_i) = \frac{\sum_{f \in D_i} E_f}{|D_i|^2} \qquad \forall D_i \in D \tag{2}$$

where $E_f$ is the residual of the feature calculated by the equation (1), $D$ is the set of all displacements coming out from the feature set to be tracked.

The displacement, with the lowest reliability factor, is selected as estimated ego-motion. By construction the displacement vector selected is the vector whose features have the minimum mean error and their number is maximum for all minimum RF displacement:

$$\mathbf{d} = D_i \quad | \quad RF(D_i) = \min_{D_i \in D} RF(D_i) \tag{3}$$

The features that have determined the displacement vector are the only features considered well tracked. All others feature are rejected.

In [11], Tommasini *et al.* reduce the detection of bad features to a problem of outlier detection based on an effective model-free rejection rule. Using this technique on a small set of features, a new problem occurs: in presence of one really worse tracked feature the value of the MAD is enough large to avoid the rejection of the single bad feature.

| | Pan Speed $\circ/sec$ | Tilt Speed $\circ/sec$ | Zoom | Autofocus |
|---|---|---|---|---|
| Seq. 1 | $10.28$ | $0.79$ | 4x | ON |
| Seq. 2 | $10.28$ | $0.79$ | 2x | ON |
| Seq. 3 | $10.28$ | $0.79$ | 1x | ON |
| Seq. 4 | $5.14$ | $0.83$ | 2x | OFF |
| Seq. 5 | $5.14$ | $0.83$ | 1x | OFF |
| Seq. 6 | $7.71$ | $0.83$ | 2x | OFF |
| Seq. 7 | $4.62$ | $0.83$ | 2x | OFF |

Table 1: Experimental sequences setups

## 4 Experimental Results

The experiments consist on testing the proposed method on outdoor sequences showing a parking area around the University of Udine. Different kinds of sequences containing all type of camera movements have been considered: pan, tilt and a combination of those two. An incremental complexity of the scenarios has been considered: scenarios in which no objects are moving, one object is moving and more objects are moving. The experiment are completed by changing the zoom, camera motion speed and the settings of intrinsic camera parameters (*e.g.*, focus and iris).

### *4.1 Camera Setup and Experiments*

The sequences used for those experiments are grabbed from a Cohu 3812 CCD camera mounted on a Pan-Tilt Unit (PTU 46-17.5). The size of the acquired images is NxM pixels, where N=384 and M=288. A Matrox METEOR-II PCI board frame-grabber has been used for image acquisition and a 1.2Ghz PC-IBM compatible has been used to run the system at the frequency of $10$ frames$/sec$ .
Seven different sequences have been considered. Table 1 shows their characteristics.
All the sequences have been acquired from the $2^{nd}$ floor of the University Campus that is about fifteen meters from the ground.
Multiple parameters have been selected to verify the algorithm efficiency. First the module displacement difference **MDD** has been considered. It represents the Euclidean difference between the estimated vector **d** and the real image displacement **rd** (the displacement that minimize the compensation error):

$$MDD = \sqrt{(d[0] - rd[0])^2 + (d[1] - rd[1])^2} \tag{4}$$

Then, the compensation error **CE** has been computed as percentage of static pixels in the change detection resulting image:

$$CE = \frac{\sum_x^N \sum_y^M B(x,y)}{N \cdot M} \tag{5}$$

where $B(x,y)$ is the binary image resulting from the change detection. These two parameters give the quality of the displacement estimation algorithm.

| | | $\mu$ | Max |
|---|---|---|---|
| TILT | MDD | 0.185 | 1 |
| | CE | 0.0016 | 0.0707 |
| PAN | MDD | 0.198 | 1 |
| | CE | 0.0022 | 0.0785 |
| PAN & TILT | MDD | 0.255 | 2 |
| | CE | 0.0051 | 0.0802 |

Table 2: Results from first scenarion

| | | $\mu$ | Max |
|---|---|---|---|
| TILT | MDD | 0.190 | 1 |
| | CE | 0.0011 | 0.0432 |
| PAN | MDD | 0.202 | 1 |
| | CE | 0.0019 | 0.0635 |
| PAN & TILT | MDD | 0.256 | 2 |
| | CE | 0.0073 | 0.0795 |

Table 3: Results from second scenario: moving objects

For each of these parameters the median $\mu$ and the maximum value $Max$ over the entire sequence has been calculated.

**A. First Scenario (6000 frames):** no moving object. This scenario contains all those results derived from pieces of sequences in which there is any moving object (Figure 2a). This is the simplest scenario since no occlusions of the features occur. The results are shown in Table 2. In all kinds of movements, the system behavior appears good. The MDD has a median value under $0.3$ that means that the displacement estimation is optimal (MDD$\approx 0$) for the majority of the frames.

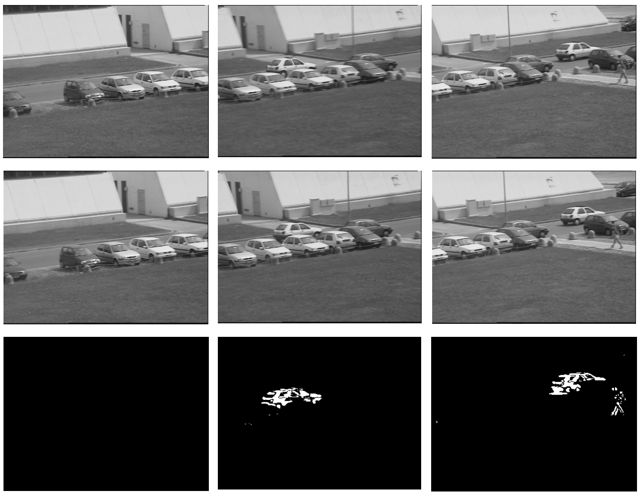
**B. Second Scenario (1400 frames):** moving objects. This part of the experimentation consists on testing the system on all the subsequences containing at least one moving object (Figure 2b and 2c). The problem complexity is increased from the first scenario since a new problem appears. The system can select, as a good feature to track, a point belonging to the moving object. Certainly, this feature has a displacement different from that of the background, so it can increase the MDD error This oblige the system to reject of the feature. The results are shown in Table 3 . The proposed rejection rule permits the presence of features belonging to the moving object only in the first frame the object appear. This feature cannot be in the set used to estimate the displacement, so it is rejected. Furthermore, the introduction of other features belonging to the object is avoided once the object is detected. So the frames in which this problem occurs are very few. As result the system behavior in this scenario is closed similar to the first. Moving objects does not involve a less accurate displacement estimation.

|  | $\mu$ | Max |
|---|---|---|
| $1^{st}$ Scenario | 1,266 | 8,246 |
| $2^{nd}$ Scenario | 3,587 | 11,180 |

Table 4: MDD values obtained using MAD technique

### *4.2 Result Comparisons*

In Table (4) is shown how the displacement estimation is performed using the MAD factor defined by Tommasini *et al.* in [11]. The experiment has been made with the same condition used for our system. The median values of the MDD parameters are always bigger than one. In the average case this technique commits at least 1 pixel error. Too bad for a good compensation.



(a) (b) (c)

Figure 2: Background compensation with displacement estimation: (a) No any object. (b) Only one object. (c)Two objects

## 5 Conclusions

In this paper we have proposed a video-based surveillance-system for image sequences acquired by a moving camera. Thanks to the displacement estimation module the system works at the frequency of $10$ frames$/sec$ on images whose size is $384 * 288$ pixels.
Results on real images demonstrate that the average displacement error ,over $7000$ frames, is under 0.25 pixel. This imply a good compensation error (CE) whose average value in less than $0.008$. The robustness of the system guarantee a good detection of mobile objects that so can be tracked and classified.

## References


[1] S. Araki, T. Matsuoka, N. Yokoya, H. Takemura, ”Real-Time Tracking of Multiple Moving Object Contours in a Moving Camera Image Sequence”, *IEICE Trans. Inf. & Syst.*, Vol. E83-D, No.7, July 2000, pp 1583-1591.

[2] R.T. Collins, A.J. Lipton, H. Fujiyoshi, and T. Kanade, ”Algorithms for Cooperative Multisensor Surveillance”, *In Proceedings of the IEEE*, October 2001, Vol. 89, No.10, pp 1456-1477.

[3] A. J. Davison, I. D. Reid, and D. W. Murray, ”The active camera as a projective pointing device”, in *Proceedings of* $6^{th}$ *British Machine Vision Conference*, Birimingham, 1995, pp 453-462.

[4] J. Gluckman and S. Nayar,”Ego-motion and omnidirectional cameras”,*In Proceedings of International Conference on Computer Vision*,Bombay, India, Jan. 3-5 1998, pp 999-1005.

[5] M. Irani, P. Anandan, *A Unified Approach to moving Object Detection in 2D and 3D Scenes.* Pattern Analysis and Machine Intelligence, Vol.20, No.6, 1998, pp. 577-589.

[6] B.D. Lucas, T. Kanade, ”An Iterative Image Registration Tecnique with an Application to stereo Vision” , *Proceedings of the* $7^{th}$ *International Joint Conference on Artificial Intelligence*, Vancouver, pp. 674–679

[7] L. Marcenaro, F. Oberti, G.L. Foresti, C. S. Regazzoni, ”Distribuited Architectures and Logical-Task Decomposition in Multimedia Surveillance Systems”, *Proceedings of the IEE*,October 2001, Vol.89, No. 10, pp 1419-1440.

[8] D. Murray, A. Basu, ”Motion Tracking with an Active Camera”, *IEEE Trans. On Pattern Analysis and Machine Intelligence*, Vol. 16, No. 5, May 1994, pp.449-459

[9] J. Shi, C. Tomasi, ”Good features to track”, *In proceedings of Computer Vision and Pattern Recognition* , 1994, pp.593-600.

[10] C. Tomasi, T. Kanade, ”Detection and tracking of point features”, *Technical report CMU-CS-91-132*, Carnegie Mellon University, Pittsburgh, PA, April 1991.

[11] T. Tommasini, A. Fusiello, E. Trucco, V. Roberto,”Making Good Features Track Better”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1998, pp. 178-183.