High Level Description of Video Surveillance Sequences

Patrick Piscaglia, Andrea Cavallaro[1], Michel Bonnet[2], and Damien Douxchamps[3]

[1] Signal Processing Laboratory, Swiss Federal Institute of Technology (EPFL)
CH-1015 Lausanne, Switzerland
Andrea.Cavallaro@epfl.ch
[2] Laboratoires d'Electronique Philips, 22 avenue Descartes, BP 15
94453 Limeil-Brévannes Cedex, France
bonnet@lep-philips.fr
[3] Telecommunications and Remote Sensing Laboratory, Catholic University of Louvain
Louvain-la-Neuve, Belgium
douxchamps@ieee.org

Abstract. One of the goals of the ACTS project MODEST is to build an automatic video-surveillance system from a sequence of digital images. The overall system can be divided into the following sub-tasks which are of great interest in the representation of images, namely the automatic segmentation of the video-surveillance sequences, and the extraction of descriptors (such as those in MPEG-7) to represent the objects in the scene and their behaviors.

1 Introduction

The European ACTS project MODEST (Multimedia Objects Descriptors Extraction from Surveillance Tapes) aims at building an automatic video-surveillance system (http://www.tele.ucl.ac.be/MODEST). The input for the system is a sequence of digital images acquired by a number of cameras installed along speedways, in tunnels, at crossroad, and so on.

The global architecture of the application can be decomposed into three main parts:

1. Segmentation of the input images, extracting video objects from the scene;
2. Description of the video objects, delivering compact and high level descriptors that ease their manipulation;
3. Reasoning based on the descriptors received. This step generates statistics, classifications, alarms, etc., and provides the user with images. The engine of this reasoning is made of Intelligent Agents.

One key particularity of the MODEST approach is its link to standardization bodies. The segmentation stage is performed in relation with the COST [3] activities. The description scheme is developed together with the development of the MPEG-7 [6]

H. Leopold and N. García (Eds.): ECMAST'99, LNCS 1629, pp. 316-331, 1999.

standard. The coding of the images transmitted to the user is MPEG-4 compliant, and the Intelligent Agents follow the evolution of the FIPA [4] standard.
The MODEST overall framework could be summarized as in Fig. 1.

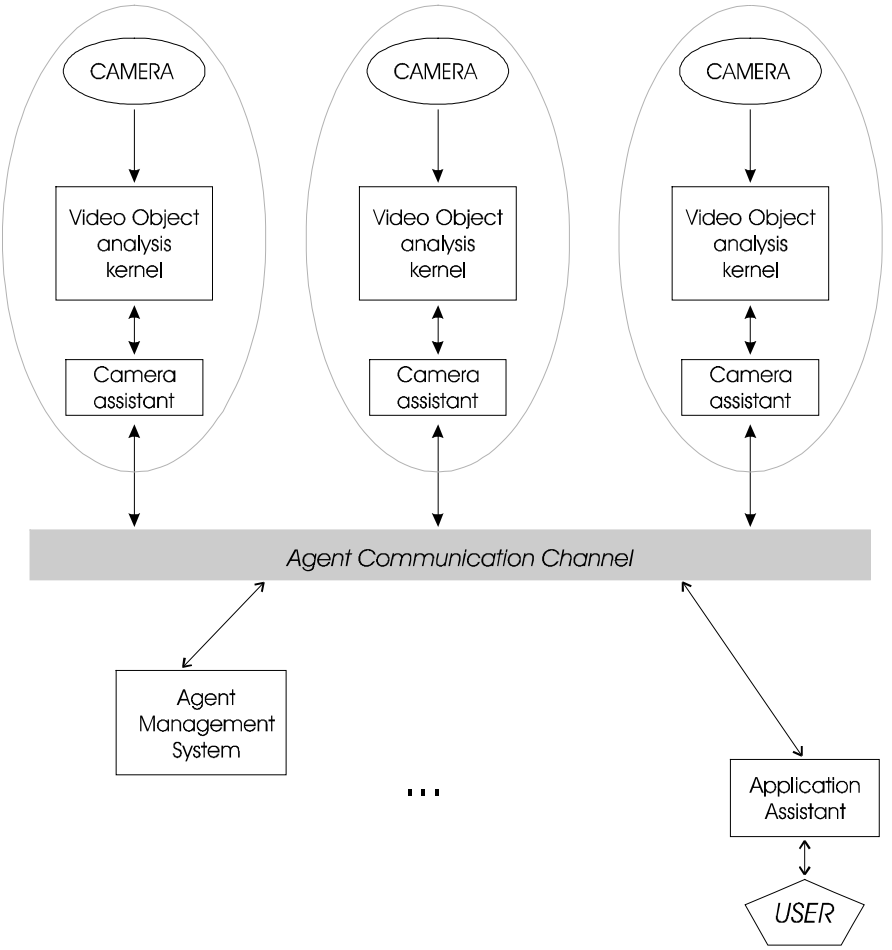

Fig. 1. MODEST overall framework

This paper will focus on the two first parts of the project. The segmentation will extract a mask for each object of the scene. The extraction uses the neighboring frame and a computed background frame to detect changes. The extraction of some features enables the software to orient the cut of the scene into meaningful objects.
The second part of the application extracts, from the object masks in combination with the original sequences, high level descriptors such as position, size, speed, orientation, color, and so on. The geometric descriptors are translated from the available 2D pixel image into the 3D world measures.

The overall tasks that the agents have to fulfill are, in the case of road monitoring:
- The detection of stopped or very slow vehicles
- The detection of vehicles driving opposite in direction.
- The classification of objects
- The tracking of a vehicle in a multiple camera environment
- The detection of vehicles driving in zigzag
- The detection of traffic jams and accidents
- The delivery of elaborated statistics, based on the results of the classification
- The computation of a global level of service (that expresses the state of obstruction of the road)
- The computation of a pollution indicator, based on some characteristics of the traffic.

The paper is structured as follows. Section 2 presents the segmentation process. Section 3 discusses the description extraction. Some visual results will be presented in Section 4. Section 5 will conclude the paper.

2 Segmentation Process

The segmentation process is the first step in the overall architecture of Modest. The aim of this step is to extract video objects from the scene. Image segmentation provides a partition of the image into a set of non-overlapping regions. The union of these regions is the entire image.

The purpose of image segmentation is to decompose the image into parts that are meaningful with respect to the particular application, in our case video surveillance. Our goal is to find regions in one frame that have changed with respect to some reference. Having a reliable reference (i.e. background) is a fundamental requisite that will be discussed in section 2.1. The detection of the regions that have changed in the image with respect to the reference is then described in section 2.2. Section 2.3 deals with the computation of the features that characterize a meaningful object mask (section 2.4).

Since this first step is based on a general segmentation approach, it does not depend on the particular application. Other proposed surveillance systems, like VIEWS [11], use parameterized 3D model-based approach able to adapt the shapes to different classes of vehicles. That choice leads to good performances but does not allow the method to be applied, for instance, to an indoor scenario.

2.1 Background Extraction

The background extraction is an iterative process that refreshes, at an instant $n+1$, the background obtained from n previous frames of the sequence with the incoming $(n+1)$ frame. All the frames of the video sequence do not need to be used in this

process: in our case, we refresh the background every two frames (The refreshment rate is therefore half the video frame rate, e.g. 3.125 Hz if the frame rate is 6.25 Hz). The accretion method uses a *blending formula* that weights regions of the incoming frame, according to their chances to belong (or not) to the background (i.e. inlier/outlier discrimination).

As shown in Fig. 2 a first error map is generated by comparing the incoming frame *(n+1)* with the current background *n*. The error map is then filtered to get rid of isolated outliers pixels. But still, this error map is *pixel-based* and detects *outliers*, mainly at the borders of moving objects, as soon as these objects are not textured enough. Moving objects are therefore only partially detected as outliers. This is the reason why *Connected Operators* [9] are used: it is a straightforward method to solve this problem that filters an image by merging its flat zones without deteriorating objects boundaries. A straightforward spatial segmentation that leads to a tree representation and detects uniform areas, in addition to a labeling process based on the error map information, enables to apply a Viterbi algorithm to prune the tree and make the decision of weighting regions (Fig. 4). This final step makes the inlier/outlier decision on whole objects by "filling the holes" efficiently. The whole process leads to a *region-based* error map.

In our case, the tree structure that defines the spatial segmentation is built from the luminance image. The resulting so-called Max-Tree grows up from dark parts of the image (the root is a region of minimal pixels value) to bright regions defining the leaves of the tree (see [10] for more details on the tree creation process). The pruning process, when applied on such a tree labelled with the error map information, eliminates more easily bright than dark moving regions. That is the reason why Connected Operators are processed twice, once on the luminance image, once on the inverse luminance image. The two resulting maps are then merged in to a final region-based weight map that will be used for blending (Fig. 3).

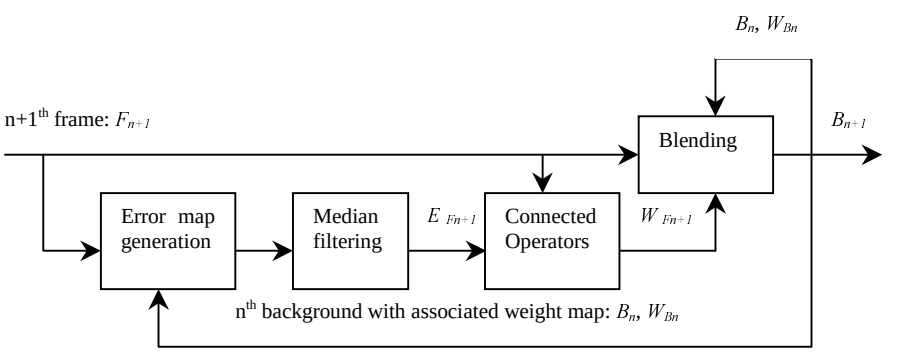

Fig. 2. Global iterative process for background extraction

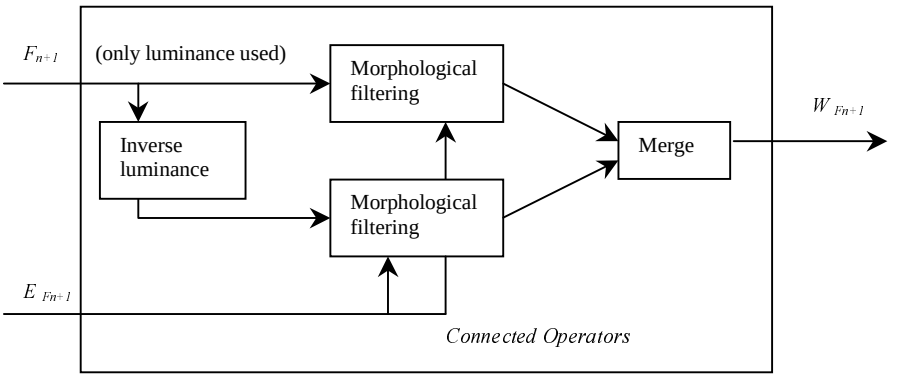

Fig. 3. Detail on the Connected Operators process (cf Fig. 2)

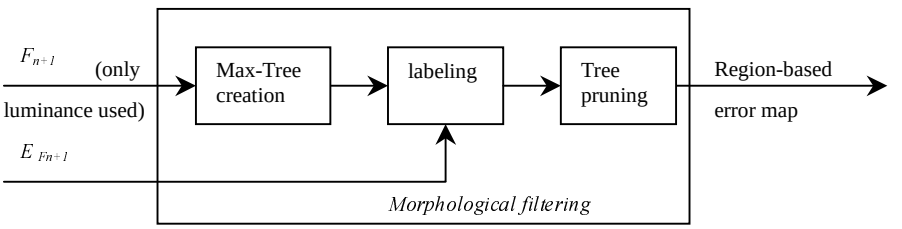

Fig. 4. Detail on the morphological filtering (cf Fig. 3)

To blend the frame *(n+1)* with the background *n*, a weighted mean formula is used to calculate the luminance and chrominance values of the new background *(n+1)*. Let $W_{F_{n+1}}$ be the previous region-based weight map and W_{B_n} the weight map of the background (accumulation of weights at each pixel). The blending formula is then given by (see Fig. 2 for notations):

$$\begin{cases} B_{n+1}(x,y) = \dfrac{W_{B_n}(x,y).B_n(x,y) + W_{F_{n+1}}(x,y).F_{n+1}(x,y)}{W_{B_n}(x,y) + W_{F_{n+1}}(x,y)} \\ W_{B_{n+1}}(x,y) = W_{B_n}(x,y) + W_{F_{n+1}}(x,y) \end{cases} \quad \textbf{(2.1)}$$

Finally, the process is iterated to obtain a background that is cleaner and cleaner at each step.
An example of extracted background is shown on Fig. 8.

2.2 Change Detection Mask

A change detection mask is a first approximation of the segmentation mask. It is a binary mask: the value 0 represents, in first estimation, a point belonging to the background while a value 1 represents a point estimated as belonging to an object.
Two techniques are combined to obtain the change detection mask: a detection based on the changing from the next and previous frames, and a detection based on the difference with an automatically extracted background image.

Change from frame detection. Two *change_from_frame detection masks* are computed, using the same technique on different input images.
The change detection detects the changing between the current frame and a reference frame. At first, the reference frame is the previous one. For the second run, the reference frame is the next frame. This method imposes a delay of one frame, but gives a much more precise mask. The estimation of the difference between the current image and the reference image is based on the value of the pixel difference between both images. The point is supposed to be a part of an object if this difference is higher than a threshold. A median filtering removes the noise and smoothes the mask.
Two masks are computed:

- A *change_from_previous_frame*, that uses the current frame and the previous frame.
- A *change_from_next_frame*, that uses the current frame and the next frame.

Change from background. The technique used to estimate the *change_from_background* mask is similar to the one used to estimate the *change_from_frame detection masks*.

Combination of change masks. The *change_from_previous_frame mask* is too big at the rear of the vehicle. The *change_from_next_frame* is too big at the front of the vehicle. A logical AND between both masks gives a *change_detection_mask* that is precise.
Large homogenous areas of the vehicles are not taken into account by this detection technique, while parts of the vehicles that have a color similar to the road are not taken into account by the *change_from_background* mask.
A logical OR between the *change_detection_mask* and the *change_from_background* mask gives the *change_detection_mask*.

2.3 Feature Extraction

In this section features chosen as significant representatives of the characteristics of the images are presented. These features constitute the input for the clustering algorithm described in 2.4. The simultaneous use of many features has the advantage of better exploiting the correlation that exists among them. In addition, it is possible to avoid the need for an iterative refinement through different stage of segmentation.

The interest of a segmentation stage after the computation of the change detection mask is related to the intrinsic limits of the information provided by the change detector. The output of the change detector, indeed, is an image representing the pixels (grouped in blobs) which are changed with respect to the reference. In other words it is a binary information. If there is only one object inside a blob in the change detection mask, this blob has a semantic meaning, i.e. it represents a car or a pedestrian. On the other hand, if there are two or more objects closed one to each other, the change detector is not able to discriminate between them. It is therefore necessary to find a strategy to overcome this limitation. In our approach we have chosen to consider the features characterizing such a blob. By clustering (see the following section) it is possible to provide the upper level (Intelligent Agents) with a richer information. This information can be exploited to separate the objects inside the same blob.

We consider two kinds of features:

- *temporal information* obtained by the computation of the motion field;
- *spatial information* considering color, texture and position.

The choice of this set of features has been driven by the results obtained after extensive simulations.

Motion information is obtained by evaluating and post-processing the optical flow. The optical flow is estimated using the algorithm proposed by Lucas and Kanade [5]. A median filtering is then performed on the resulting motion field in order to avoid the effect of spurious vectors.

Among the different options for choosing the *color* space, the YUV coordinates have been selected since they allow a separate processing of the luminance (Y) information. The color information undergoes a post-processing stage (by a median filtering) that reduces the noise while preserving the edges. The use of the spatial coordinates (*position* feature) of each pixel helps in increasing the level of spatial coherence of the segmentation and thus the compactness of the resulting regions. A *texture* feature is finally taken into account. This value characterizes the amount of the texture in a neighborhood of the pixel and is defined as standard deviation of the gray level over a 3x3 window. The result is then post-processed to reduce the estimation error due to the presence of edges.

2.4 Object Mask

The extraction of regions is based on several features computed from the image as described in the previous section. Given the change detection mask (see section 2.2) and the set of features, we take into account only the pixels that have been classified as changed. The Fuzzy C Means algorithm [1] is used to identify regions that are homogeneous in the feature space. As said earlier, these regions can be grouped together by the Intelligent Agents to form objects.

3 Descriptions Extraction

The descriptions extracted by this section of the algorithm intend to be MPEG-7 compliant. At the time of this writing, MPEG-7 standard is not finalized yet. A Call for Proposals (CFP) [2] has taken place in February 1999, to which four description schemes have been submitted by MODEST, namely the TimeStamp, GlobalSceneParameters, Trajectory and XenoObject description schemes.
A geometric measure alone is not useful in the 2D-pixel image to describe accurately real dimensions in the scene. For instance, if a vehicle moves away from the camera, its size expressed in pixels decreases, while its real size of course does not change. A conversion from the 2D data to 3D world measures is then mandatory to have meaningful descriptions, which is why there is a lot of work in the field of 2D-pixel to 2D-meters coordinates mapping [8]. This conversion is only achievable if the camera has previously been calibrated, which is first discussed. The originality of our approach is the use of full 3D coordinates to describe the scene: not only we locate the object in real-world coordinates the ground plane, but we also determine its height and 3D-shape. We also use a non-model-based approach for generality, which differs from other research in the field [11]. The different descriptors are described in the following subsection, while the future work that the project is about to perform on the descriptions, is drafted in the last subsection.

3.1 Camera Calibration

The preliminary step before three-dimensional reconstruction of the scene is the camera calibration. The goal of this process is to compute the parameters of the central projection of the world on the image plane of the camera. One should note that this transform will not be enough to compute 3D coordinates: it is only able to give us the equation of the line (in world coordinates) passing through the image point and the focal center of the camera. Such a line is called a 'ray', and the process that will remove the last unknown (i.e. change the ray to a single 3D point) is discussed in the 2D-3D conversion section.
This world to image transformation (a central projection) is approximately affine, so that we need to find 12 parameters: 9 for the rotation matrix (the camera orientation) and 3 for the translation vector (the camera location in world coordinates). The equation that links 3D to 2D coordinates is then:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \frac{1}{R}\left[\begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}\right], \tag{3.1}$$

where *(X,Y,Z)* are the world coordinates, (x_i,y_i) the image coordinates, *R* the rotation matrix and *t* the translation vector. This equation can be used to find the *R* matrix and *t* vector providing that both world coordinates and image coordinates are available for some points, called calibration points. If such calibration points are delivered to the system, all the parameters are then the solution of a simple linear system.
When *R*, *t* and (x_i,y_i) are known, the directing vector of the ray associated with a image point can be computed, the translating vector of the ray being always *t*.

3.2 2D to 3D Conversion

The classic way to convert two-dimensional pixel coordinates into three-dimensional world coordinates would be to use stereovision. However, in our case, there is only one camera for each observed scene.
In general, three-dimensional reconstruction is possible if several spatially different views of the same object can be obtained. In the case of a single camera, different views can be obtained in the way that the objects will move from one frame to another. The movement will cause the perspective to change, yielding the different views for the object.
The complete process consists in finding interesting points on the moving objects, matching them between two frames and deducing the three-dimensional coordinates from the two different locations of the feature point.
The feature detector used is the Moravec [7] operator. The detection of significant features is done by thresholding the resulting feature image and matching its neighborhood in the original image with a region in the next frame (special BMA algorithm, directed by an interest map). Inverting the first and the second frame in this process yields two sets of pairs of points. We then only select the pairs that are the same in the two sets. The selection of the best pairs further downsizes the set.
The three-dimensional reconstruction requires several hypotheses. Objects must be rigid and in motion (see above). The objects should also behave following a plane movement, i.e. each feature point has the same height in each frame. This height conservation hypothesis is combined with the supposition of the existence of points located on the ground to yield a set of equations determining the free parameter of the rays, and hence the three-dimensional points.
If we suppose Pg on the ground (see Fig. 5) and P above it, the true X-Y location Pr of P will be a function of the projection Pp of P on the ground plane. If we have two frames for this object, the distance conservation between the frames will yield a determined system for the Pr location of the three-dimensional point P:

perspective equations for t_1 and t_2: $P_r = (1 - P(z)/t(z))P_r$ **(3.2)**

rigidity equation: $\| P_r(t_1) - P_g(t_1) \| = \| P_r(t_2) - P_g(t_2) \|$ **(3.3)**

where P(z) is the altitude of point P and t(z) the altitude of the camera.

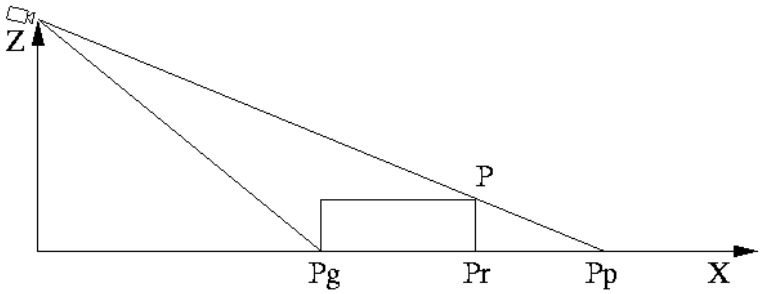

Fig. 5. Three-dimensional reconstruction: projected view of the 3D scenario

Results from this process are shown on Fig. 6. The bounding boxes show the reconstructed 3D limits of the vehicle, while the tails are the trajectories of the projection of the center of mass of the vehicle on the ground.

Fig. 6. Vehicles bounding boxes and trajectories

3.3 Descriptions

The descriptions are high level ways of representing the information contained in the scene. Once the objects are extracted, they can be characterized by their dimension, their speed, their position, their orientation, their color and their shape. The global

conditions on the scene are also extracted to deliver to the reasoning agents a complete information.
The description schemes proposed at the MPEG-7 CFP, and mentioned at the beginning of section 3 are made up with a subset of the following descriptors.

Dimension. The descriptor for the dimension of the vehicle gives the 3D-dimension of the object along its main and perpendicular axis. Three values are delivered: length, width and height.
After the 2D to 3D conversion, the X dimension of the objects is the distance between the minimum and maximum X coordinates. The Y and Z dimensions are computed with the same rule.

Speed. The descriptor for the speed gives the (x,y,z) speed of an object from the previous frame. Three values are delivered: x,y,z motion.
After the 2D to 3D conversion, the speed vector is the difference between the position of the center of mass of the object at the current frame and at the previous frame, divided by the time separating both frames.

Position. The descriptor for the position gives the (x,y,z) position of the mass center of the object. The (x,y,z) position is delivered.
After the 2D to 3D conversion, the position is the 3D-position of the center of mass of the object.

Orientation. The descriptor for the orientation gives the orientation of the main axis of an object, defined as the angle between the main axis and the horizontal axis (using the standard trigonometric referential).
After the 2D to 3D conversion, the orientation of the vehicle is the angle given by

$$angle = \frac{1}{2}\arctan\left(\frac{m_{11}}{m_{20}}\right) \tag{3.4}$$

where m_{11} stands for the first order xy-momentum and m_{20} stands for the second order x-momentum.

Color. The descriptor for the color gives the colors of an object as a set of (H,S,V) values (one triplet of value for each dominant color of the object).

Shape. A lot of possibilities exist for describing the shape of an object. The simplicity, efficiency, speed and relevancy for higher levels of several algorithms have been under study. The contour has finally been chosen.
The description scheme that gives the contour of the object is a set of (x,y,z) points representing the approximation of the contour of the object if joined by a straight line.

Global conditions. The global conditions on the scene can be expressed by several descriptors or description schemes:

- The global color gives the global H, S and V values of the scene, and their variance. The global color can vary for example with the color calibration of the camera.
- The global motion of the camera gives the global motion (or tilt) of the camera. Despite the fact that the camera is supposed to be fixed, small motions or tilts can occur due to the wind, abnormal vibrations…
- The number of objects identified in the scene.

3.4 Future Work

The coding of the descriptors and description schemes is currently under study. Two coding scheme are developed:

- A textual coding scheme that represents the data as a text. An example of textual coding is given in section 4.
- An efficient coding. The aim is to obtain a bytecode as small as possible to describe the descriptors and description schemes. In the Modest application, the segmentation and description tools are located closed to the camera. The reasoning can be located in the central dispatching. The descriptions must be sent as fast as possible, on communication lines as cheap as possible.

4 Results

This section presents some results of the algorithms described in the sections 2 and 3. Fig. 7 presents a few snapshots of some of the sequences.

Fig. 7. Snapshots from the original sequences

Fig. 8 shows the background that is automatically extracted from the sequences.

Fig. 8. Background image

For the first snapshot, Fig.9a shows the Change Detection Mask obtained with the previous frame and Fig. 9b shows the Change From Background (CFB) mask. The combination of the two CDM and the CFB mask is represented in Fig. 9c to form, after segmentation and morphological filtering, the object masks presented in Fig. 10 for the two snapshots.

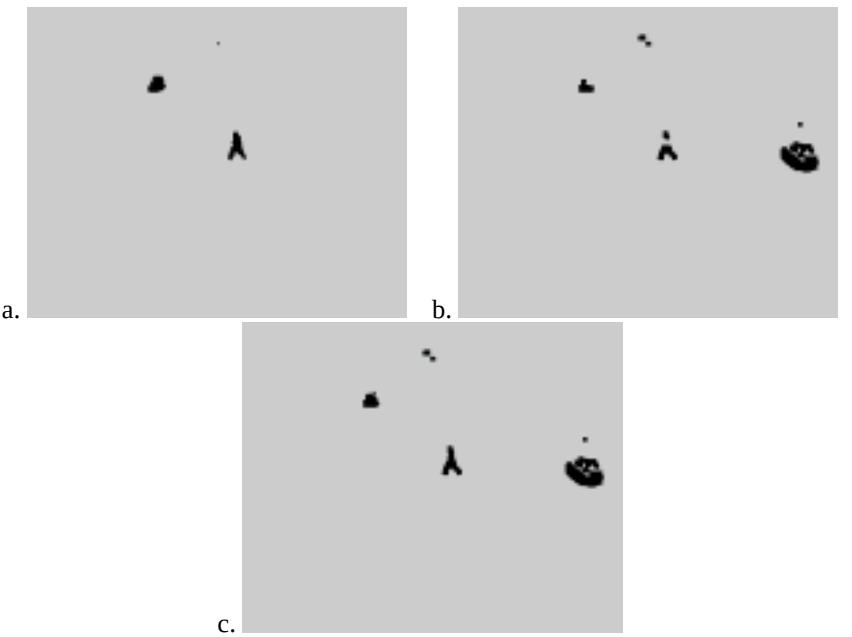

Fig. 9. Change Detection (i, i-1), Change From Background, and final CD masks

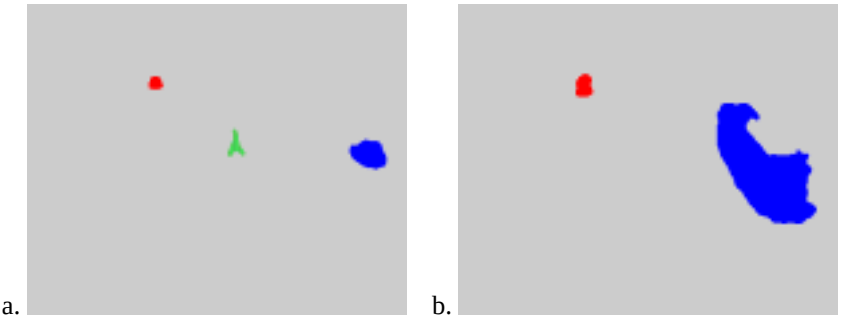

Fig. 10. Object Masks

Fig. 11 displays the real extracted objects. Objects that are very far in the field are not taken into account because they are too small. One can see that the segmentation is quite accurate for most of the vehicles of peoples, except for the load of the truck, that is a large homogeneous zone, with a color close to the background color. The 2D to 3D conversion will nevertheless find enough correct points to obtain the 3D shape of the truck.

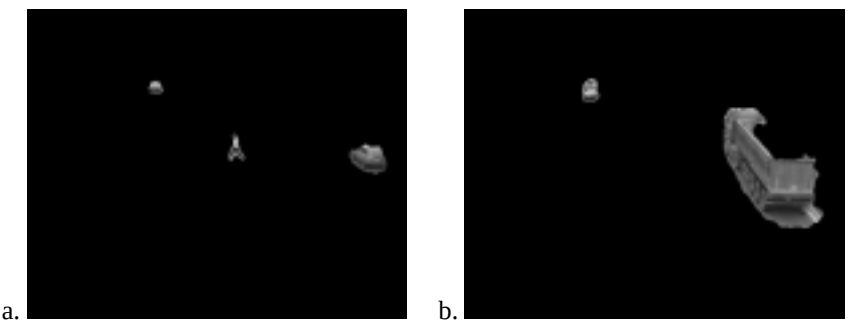

Fig. 11. Visual representation of the extracted objects

The second part of the algorithm computes the descriptions extracted from the object masks, in combination with the original images. The textual representation of the descriptors and description schemes is presented.

```
:object (
     :id 1
```

```
    :size (   // Expressed in meters
        :length (:value 4.23 :confidence 0.93)
        :width (:value 1.56 :confidence 0.87)
        :height (:value 1.42 :confidence 0.81)
    )
    :trajectory (
        :position (    // Expressed in meters
            :X (:value 40.53 :confidence 0.91)
            :Y (:value –6.20 :confidence 0.86)
            :Z (:value 0.76 :confidence 0.65)
        )
        :speed ( // Expressed in meters/second
            :X (:value 29.12 :confidence 0.91)
            :Y (:value 0.11 :confidence 0.84)
            :Z (:value 0.02 :confidence 0.21)
        )
        :orientation ( // Expressed in radian
            :value 1.59
            :confidence 0.94
        )
    )
    :color (
    . . .
    )
)
```

5 Conclusion

This paper has presented a representation scheme of images in two steps. The first step consists of an efficient segmentation of images in order to obtain object masks that ease the extraction of a high level description of the objects present on the scene. The second step consists of an extraction of high level descriptors and description schemes from the above object masks. The descriptions have been submitted to the MPEG-7 standardization group.

We have shown in the paper different techniques combined to obtain the object masks, from the previous, the current, the next frame, and from the background image. We have also shown the way to extract descriptions for the objects, containing the size, the position, the orientation, the speed, the color, and so on, and some trails for the coding/decoding of the descriptions for their transmission.
A third step, which is out of the scope of this paper, has been briefly presented. It describes the way the descriptions are used by the application developed in the framework of the ACTS project MODEST.

References

1. R. Castagno, T. Ebrahimi, M. Kunt. "Video Segmentation based on Multiple Features for Interactive Multimedia Applications". In IEEE Transactions on Circuits and System for Video Technology, Vol.8, No.5, pp.562-571, September 1998
2. ISO/IEC JTC1/SC29/WG11 N2469 "Call For Proposals for MPEG-7 Technology", October 1998, Atlantic City, USA
3. The European COST211quat Group home page (http://www.teltec.dcu.ie/cost211)
4. The Foundation for Intelligent Physical Agents (FIPA) home page (http://www.fipa.org)
5. B. Lucas, T. Kanade. "An iterative image registration technique with an application to stereo vision". In Proceedings of 7th International Joint Conference on Artificial Intelligence, pp. 674-679, 1981
6. The Moving Picture Expert Group (MPEG) home page (http://drogo.cselt.stet.it/mpeg)
7. H.P. Moravec, “Towards automatic visual obstacle avoidance”, Proc. 5th Int. Joint Conf. Artificial Intell., vol. 2, pp. 584, August 1977
8. C.S. Regazzoni, G. Fabbri, G. Vernazza, “Advanced Video-Based Surveillance Systems”, Kluwer Academic Publishers, 1999
9. P. Salembier, J. Serra, “Flat zones Filtering, Connected Operators, and Filters by reconstruction”, IEEE Trans. On Image Processing, vol. 4, No 8, pp. 1153-1160, August 1995
10. P. Salembier, A. Oliveras, L. Garrido, “Anti-extensive Connected Operators for Image and Sequence Processing”, IEEE Trans. On Image Processing, vol. 7, No 4, pp. 555-570, April 1998
11. The VIEWS project home-page (http://www.cvg.cs.rdg.ac.uk/views/home.html)
12. COST XM document (http://www.teltec.dcu.ie/cost211)