Bayesian Fusion of Visible Cameras for Behaviour Recognition

Julien Ros[1] Kamel Mekhnacha[1]

[1] Probayes SAS

345, rue Lavoisier - Inovallée
38330 Montbonnot – FRANCE

{julien.ros, kamel.mekhnacha}@probayes.com

Abstract

The utilisation of several cameras to monitor human activity in a large space is essential due to the important field of view to be covered and the possible cluttered environment. The interpretation of this high number of data requires fast and powerful fusion algorithms in order to make easier the next human or computer work. In this paper the utilisation of a probabilistic occupancy map is proposed to fuse videos coming from different cameras. By estimating the occupancy and the velocity of each spatial cell representing the environment and obtained thanks to a background substraction algorithm, it is shown that human can be efficiently tracked. The tracking information is finally successfully used by a bayesian filter to recognise low level pedestrian behaviour such as standing, walking and running.

Keywords

Bayesian Occupancy Filter, visible camera fusion, behaviour recognition.

1 Introduction

Nowadays, employing video cameras to monitor a place has become very popular. Cameras can be used in home applications for power saving while facilitating the user everyday life; in supermarkets, to increase the average individual sales by bringing some interactivity and in the security domain to detect abnormal situations. Considering the security market in the United Kingdom and according to a study of urbaneye[1], there were approximately 4,200,000 cameras in 2002 in UK which represents one camera for every 14 people. Efficient systems used to facilitate the interpretation of this high quantity of information should thus be found. Indeed, multiple sensors could provide more reliable and robust information about the environment. As a consequence, these applications should be able to fuse different inputs provided by different sensors in order to make decisions about the current situation monitored. In practice, a videosurveillance application often requires four main steps (see Figure 1): human detection, sensor fusion, human tracking, and human behaviour recognition. This paper deals with all steps. Due to the sensor measurements which are noisy, it's seldom possible to construct an exact representation of the environment monitored by the different sensors. Thus, probabilistic approaches are usually used to cope with this problem and especially those using occupancy grids [1, 2, 3, 4, 5].

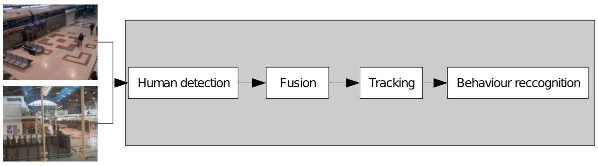

Figure 1 – *Multi-camera Human Behaviour Recognition System Architecture*

Among them, the Bayesian Occupancy Filter (BOF) introduced in [6], improved in [7], has already been used to perform videosurveillance task by fusing visible and infrared cameras in [8]. For this purpose, the BOF employs deeply the Bayesian approach in order to perform sensor fusion for the occupancy map estimation. It proposes to use a global filtering equation to estimate both the occupancy and the velocity of a given grid cell. The occupancy map is then given as input to a clustering algorithm to extract objects tracked in the next step.

Once human are correctly tracked, it is possible to perform human behaviour recognition and for this purpose, Bayesian approach are generally employed [9, 10, 11]. However, they often require a precise limb description of the human tracked difficult to obtain in large area surveillance or they are based on a training phase difficult to handle when the training set is too small. In this paper, we show that the position and velocity tracking information provided by the BOF are enough to employ a Bayesian filter to detect basic behaviour such as standing, walking and running.

The paper is organized as follows. Section 2 presents how the information provided by the human detection algorithm are fused into a single Bayesian occupancy map in order to

[1] http://www.urbaneye.net/results/ue_wp6.pdf

return track associated to each pedestrian monitored. Section 3 describes the Bayesian filtering process employed to detect low level behaviours. Experiments are presented in Section 4 and finally Section 5 concludes the paper.

2 Bayesian Occupancy Filter (BOF)

The Bayesian Occupancy Filter (BOF) is represented as a two-dimensional grid-based decomposition of the environment. Each cell of the grid contains two probability distributions: (i) the probability distribution over the occupancy of the cell (ii) and the probability distribution over its velocity. Given a set of input sensor readings, the BOF algorithm allows to update the occupancy/velocity estimates of each grid cell.
Figure 2 shows an example of BOF output using a computer vision car detector.

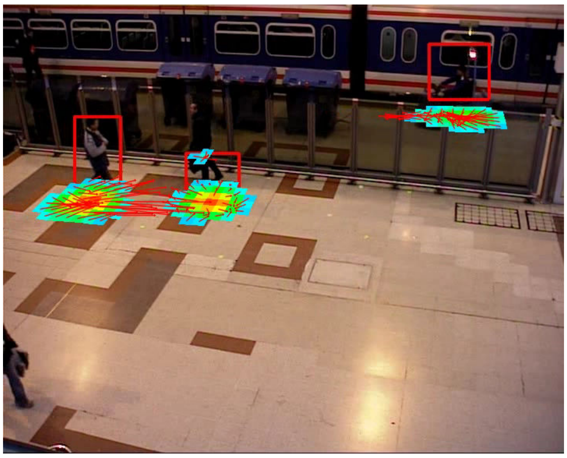

Figure 2 – *Example of BOF output using a computer vision pedestrian detector as input (red boxes). The BOF output is projected back on the image. It represents a grid of occupancy probability (blue-to-red mapped color) and the mean velocity (red arrows) estimates.*

A more detailed description of the BOF framework should be found in [12]. The BOF model is shown graphically in Fig. 3 and is described as follows:

2.1 Variables

For a given cell having $c \in \mathcal{Y}$ as index in the grid, let:

- $A_c^t \in \mathcal{A}_c \subset \mathcal{Y}$ represents each possible antecedent of cell c over all the cells in the grid domain $\mathcal{Y}$. The set of antecedent cells of cell c is denoted by $\mathcal{A}_c$ and is defined as a neighbourhood of the cell c.

- $A_c^{t-1} \in \mathcal{A}_c \subset \mathcal{Y}$ the same as A_c^t but for the previous time step.

- $O_c^t \in \mathcal{O} \equiv \{0, 1\}$ is a boolean variable representing the state of the cell in terms of occupancy at time t, either $[O_c = 1]$ if occupied, $[O_c = 0]$ if empty. Given the independency hypothesis, the occupancy of each cell at time t is considered apart from the occupancy of its neighbouring cells at time t.

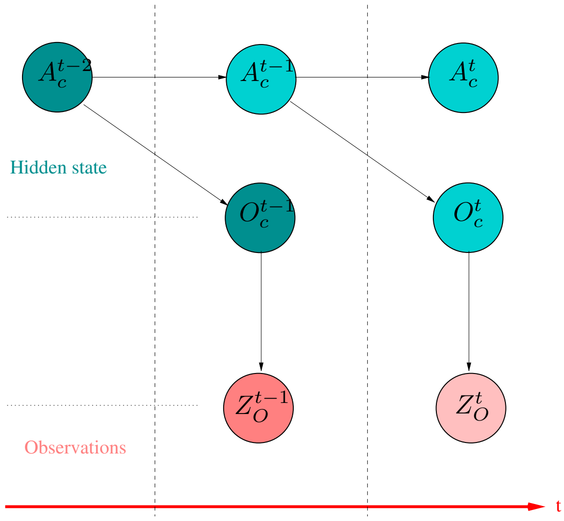

Figure 3 – *The Dynamic Bayesian Network corresponding to the BOF model for each grid cell c. Here, we suppose that only occupancy sensors are available.*

- $Z_i^t \in \mathcal{Z}, 1 \le i \le S \in \mathbb{N}$, is a generic notation for measurements yielded by each sensor i, considering a total of S sensors yielding a measurement at the considered time instant.

2.2 Joint distribution factors

The following expression gives the decomposition of the joint distribution of the relevant variables according to Bayes' rule and dependency assumptions:

$$P(A_c^{t-1}\, A_c^t\, O_c^t\, Z_1^t \cdots Z_S^t) =$$
$$P(A_c^{t-1}) P(A_c^t \mid A_c^{t-1}) P(O_c^t \mid A_c^{t-1}) \prod_{i=1}^{S} P(Z_i^t \mid A_c^t\, O_c^t) \quad (1)$$

The parametric form and semantics of each component of the joint decomposition are as follows:

- $P(A_c^{t-1})$ is the probability for a given neighbouring cell A_c to be the antecedent of c at time $t-1$. In order to represent the fact that cell c is *a priori* equally reachable from all possible antecedent cells in the considered neighbourhood, this probability table is initialized as uniform and is update in each time step.

- $P(A_c^t \mid A_c^{t-1})$ is the distribution over antecedents at time t given the antecedent of cell c at $t-1$. It represents the prediction (dynamic) model over velocity. If we assume a perfect *constant velocity hypothesis* between the two time frames $t-1$ and t, this distribution is simply:

$$P(A_c^t \mid A_c^{t-1}) = P(A^{t-1}_{A_c^{t-1}}).$$

In other words, the predicted probability is simply the probability at the preceding time instant for the antecedent at $t-1$.

Considering imperfect *constant velocity hypothesis* is possible by introducing the predicate $E \in \{0,1\} \equiv$ "There was an erroneous prediction", and assuming a probability $P(E) = \epsilon$. This value is a parameter of the system and corresponds of the probability of not respecting the *constant velocity hypothesis*. We have:

- $P(A_c^t \mid A_c^{t-1}\ \neg E) = P(A^{t-1}_{A_c^{t-1}})$,
- $P(A_c^t \mid A_c^{t-1}\ E) = \mathcal{U}(A_c^t)$,

where $\mathcal{U}(A_c^t)$ denotes a uniform distribution on A_c^t to say that all possible antecedents have the same probability when *constant velocity hypothesis* is not respected. Thus, $P(A_c^t \mid A_c^{t-1})$ may be written as a mixture:

$$\begin{aligned} &P(A_c^t \mid A_c^{t-1}) = \\ &\quad P(\neg E)P(A_c^t \mid A_c^{t-1}\ \neg E) + P(E)P(A_c^t \mid A_c^{t-1}\ E). \end{aligned}$$

Which leads to:

$$\begin{aligned} P(A_c^t \mid A_c^{t-1}) &= (1-\epsilon)P(A^{t-1}_{A_c^{t-1}}) + \epsilon\,\mathcal{U}(A_c^t) \\ &= (1-\epsilon)P(A^{t-1}_{A_c^{t-1}}) + \epsilon/\|\mathcal{A}_c\|, \end{aligned}$$

where $\|\mathcal{A}_c\|$ is the cardinality of the considered antecedents set $\mathcal{A}_c$.

- $P(O_c^t \mid A_c^{t-1})$ is the distribution over occupancy given the antecedent of cell c at $t-1$. It represents the prediction (dynamic) model over occupancy. Similarly to $P(A_c^t \mid A_c^{t-1})$, the term $P(O_c^t \mid A_c^{t-1})$ may be written as a mixture:

$$\begin{aligned} P(O_c^t \mid A_c^{t-1}) &= (1-\epsilon)P(O^{t-1}_{A_c^{t-1}}) + \epsilon\,\mathcal{U}(O_c^t) \\ &= (1-\epsilon)P(O^{t-1}_{A_c^{t-1}}) + \epsilon/2. \end{aligned}$$

- $P(Z_i^t \mid A_c^t\, O_c^t)$ is the *direct model* for sensor i. It yields the probability of a measurement given the occupancy O_c^t and the antecedent (velocity) A_c^t of cell c. Measurements for all sensors are assumed to have been taken *independently from each other*. For sensors providing measurements depending exclusively of occupancy, this distribution can be written as $P(Z_i^t \mid O_c^t)$. In the same manner, for sensors providing measurements depending exclusively of velocity, this distribution can be written as $P(Z_i^t \mid A_c^t)$.

2.3 Occupancy and velocity estimation using the BOF model

At each time step, the estimation of the occupancy and velocity of a cell is answered through Bayesian inference on the model given in Equation (1). This inference leads to a Bayesian filtering process (Fig. 4).

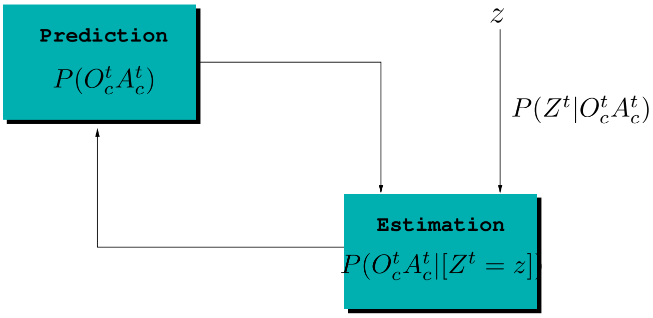

Figure 4 – *Bayesian filtering in the estimation of occupancy and velocity distribution in the BOF grids.*

2.4 A Gaussian Image Model Associated To Visible Cameras

It is generally assumed that the first step involved in a video surveillance application is the human detection. In this paper, we consider system where the cameras involved are static, humans are thus considered as moving regions in front of a relatively static background and a simple and inexpensive method to perform the detection is to use a background subtraction algorithm including foreground discrimination and blob segmentation.
To use these blobs as input to the BOF, a sensor model $P(Z_i^t \mid O_c^t)$ that takes as input the human bounding boxes is employed. This sensor model considers as sensor observations the bounding boxes and project them on the grid thanks to the calibration information in a Gaussian way as shown in Figure 5.

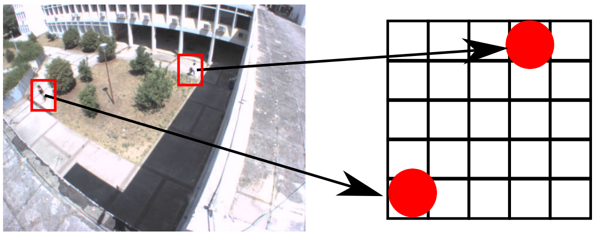

Figure 5 – *Insertion of the Blobs in the Bayesian Occupancy Grid.*

In fact this model considers that each detected bounding box represents an object on the floor (i.e. that the lower part of the bounding box corresponds to the floor). Thus by using the calibration matrix, it generates a Gaussian occupancy probability centred on this object and whose standard deviation is proportional to the bounding box width. The size (covariance matrix) of the Gaussian represents the localisation error.

2.5 Human Tracking From the BOF Output

As previously emphasized, the BOF allows to represent the surrounding environment by a map whose cells have an occupancy and velocity distribution. However, we are more interested, in this paper, in knowing the number of

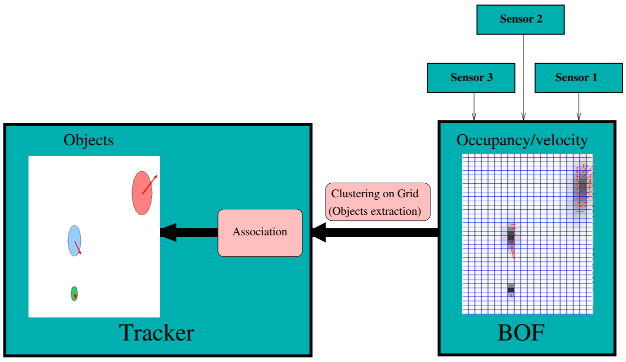

Figure 6 – *Sensing/Tracking System Architecture.*

people and their movement in the monitored area. People should thus be extracted from the BOF output and the people found need to be tracked.

For this purpose, the occupancy/velocity map is given as input to a clustering-tracking algorithm to extract objects tracked in the next step. The principle of this method is presented on Fig. 6 and a more detailed description of this clustering-tracking part could be found in [13].

It is important to note that thanks to the clustering procedure postponed at the end of the fusion process, the BOF has the advantage of not being based on an object-based representation. It allows a complexity reduction of the data association stage which could be encountered in a cluttered environment. The algorithm is highly parallelisable and can thus be used in real time to perform sensor fusion.

3 Behaviour Recognition From BOF Output

The objective of this section is to propose a filtering scheme for classifying the current motion mode of a given individual. The idea is to use the velocity information provided by the tracking system for estimating the current motion mode among three discrete hypotheses $\{Standing, Walking, Running\}$.

Our approach is based on a Bayesian filtering scheme in order to recursively update the belief about the motion mode. At each time index, the filter uses the speed observation (a Gaussian estimate) provided by the tracker for updating the belief table over the three possible hypotheses $\{Standing, Walking, Running\}$.

In order the describe the model, let's define the following variables associated to a track (a person):

- B^t: The current behaviour (at time index t), $B^t \in \{Standing, Walking, Running\}$.
- B^{t-1}: The previous behaviour (at time index $t-1$), $B^{t-1} \in \{Standing, Walking, Running\}$.
- S^t: The current speed (at time index t), $S^t \in \mathbb{R}$.
- S^t_{obs}: The observed speed (at time index t), $S^t_{obs} \in \mathbb{R}$.

The joint distribution corresponding to the proposed filter is:

$$P(B^{t-1}\ B^t\ S^t\ S_{obs}) = \\ P(B^{t-1})P(B^t \mid B^{t-1})P(S^t \mid B^t)P(S^t_{obs} \mid S^t),$$

in which:

- $P(B^{t-1})$: The probability distribution corresponding to the estimation of the behaviour at the previous time index $t-1$.
- $P(B^t \mid B^{t-1})$: The prediction model providing the transition probabilities from a given mode to another. $P(B^t \mid B^{t-1})$ is the probability of switching to mode $P(B^t)$ (at time t) given the previous mode is B^{t-1} (at time $t-1$). This conditional distribution has been set by hand to the matrix in Tab. 1.

	Standing	Walking	Running
Standing	0.90	0.05	0.05
Walking	0.05	0.90	0.05
Running	0.05	0.05	0.90

Table 1 – *Transition matrix.*

- $P(S^t \mid B^t)$: The observation model providing the distribution over the speed given the actual mode. It's supposed to be Gaussian:

$$P(S^t \mid B^t) = \mathcal{N}(S^t; \mu(B^t), \sigma(B^t)),$$

in which the parameters have been set by hand to the matrix in Tab. 2. However, it is important to note that a learning scheme (E.M. algorithm) can be employed to set these parameters.

	Standing	Walking	Running
$\mu(B^t)$	0.0	0.5	3.0
$\sigma(B^t)$	0.1	0.2	0.5

Table 2 – $P(S^t \mid B^t)$ *parameters. The unit is meter per second.*

- $P(S^t_{obs} \mid S^t)$: The observation error model. It's assumed Gaussian:

$$P(S^t_{obs} \mid S^t) = \mathcal{N}(S^t_{obs}; S^t, \sigma^t_{obs}),$$

in which σ^t_{obs} is the standard deviation associated to the estimated speed S^t_{obs} returned by the tracker.

4 Experimental Results

The proposed fusion scheme has been applied to the Prometheus European project data set (http://www.prometheus-fp7.eu/) and especially to the "ATM" scenario.

This scenario focuses on security around an automated teller machine. The corresponding data has been recorded in a wide outdoor area with two visible cameras (1024x768, 15fps).

All these sensors have been calibrated using the Camera Calibration Toolbox available at (http://www.vision.caltech.edu/bouguetj/calib_doc).

To detect moving blobs, a classical background subtraction algorithm is used [14]. For this purpose, a modified version of the efficient implementation provided by the OPENCV library using three adaptive Gaussian to model the background is employed. However, it has been improved to not update background components where blobs were detected at the previous step. It allows to continue to detect standing people even if they do not move for a long time.

The tracking and behaviour recognition results are shown in Figure 7 where people behaviours are displayed in red.

First, it can be seen that the tracking is very robust even when people are following each other resulting in a severe occlusion problem. The different points of view associated to the used sensors and the fusion process associated allow to reach this level of robustness.

Second, it is easy to see that the behaviours are correctly estimated because the system can efficiently discriminate standing, walking and running people thanks to the utilisation of the Bayesian filter.

5 Conclusions and Discussions

In this paper, we used the Bayesian Occupancy Filter framework to fuse the information provided by different cameras monitoring the same field of view. In the videos, pedestrians were detected thanks to a classical background subtraction algorithm whose results were given as input to the BOF. The BOF allows the computation of a grid storing both occupancy and velocity of each cell. This output was then clustered in order to obtain an object based representation of the environment allowing the pedestrian tracking and low level behaviour recognition thanks to a Bayesian Filtering process

The results show that this system can be efficiently used to resolve occlusions that often occur in a videosurveillance application. Moreover, the small computing times allow to envisage its integration in a commercial system.

However, the behaviours detected are always too simple and we are currently working on using a concurrent HMMs-based model in order to recognise high level behaviour using the motion information provided by the human tracks.

References

[1] Elfes A.. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[2] Thrun S., Fox D., et Burgard W.. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31:29–53, 1998.

[3] Fleuret F., Berclaz J., Lengagne R., et Fua P.. Multi-camera people tracking with a probabilistic occupancy map. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 30(2):267–283, 2007.

[4] Ferreira J.F., Bessière P., Mekhnacha K., Lobo J., Dias J., et Laugier C.. Bayesian models for multimodal perception of 3d structure and motion. Dans *Proceedings of the International Conference on Cognitive Systems (CogSys 2008)*, April 2008.

[5] Beymer D.. Person counting using stereo. Dans *HUMO '00: Proceedings of the Workshop on Human Motion (HUMO'00)*, page 127, 2000.

[6] Coué C., Fraichard T., Bessière P., et Mazer E.. Multi-sensor data fusion using bayesian programming : an automotive application. Dans *Proceedings of the IEEE-RSJ International Conference on Intelligent Robots and Systems*, 2002.

[7] Tay C., Mekhnacha K., Chen C., Yguel M., et Laugier C.. An efficient formulation of the bayesian occupation filter for target tracking in dynamic environments. *International Journal Of Autonomous Vehicles*, 6(1/2):155–171, 2008.

[8] Ros J. et Mekhnacha K.. Multi-sensor human tracking with the bayesian occupancy filter. Dans *Proceedings of the 16th International Conference on Digital Signal Processing (DSP 2009)*, pages 1–8, Santorini, Greece, 2009.

[9] Oliver N.M., Rosario B., et Pentland A.P.. A bayesian computer vision system for modeling human interactions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):831–843, 2000.

[10] Nascimento J.C., Figueiredo M.A.T., et Marques J.S.. Segmentation and classification of human activities. Dans *Proceedings of HAREM International Workshop on Human Activity Recognition and Modelling*, 2005.

[11] Devlaeminck R.. *Human Motion Tracking With Multiple Cameras Using a Probabilistic Framework for Posture Estimation*. Thèse de doctorat, School of Electrical and Computer Engineering, Purdue University, 2006.

[12] Mekhnacha K., Mao Y., Raulo D., et Laugier C.. Bayesian occupancy filter based 'fast clustering-tracking' algorithm. Dans *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, 2008.

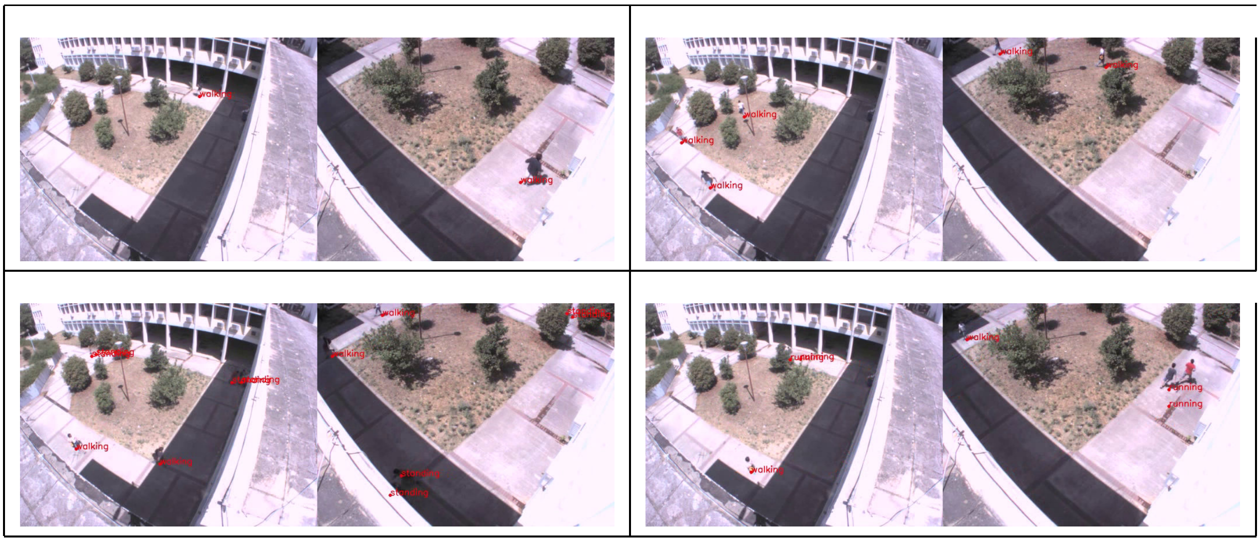

Figure 7 – *Tracking and Behaviour Recognition Results Using Two Visible Cameras.*

[13] Mekhnacha K., Mao Y., Raulo D., et Laugier C.. The 'Fast Clustering-Tracking' algorithm in the Bayesian Occupancy Filter framewor. Dans Springer Berlin Heidelberg, éditeur, *Multisensor Fusion and Integration for Intelligent Systems*, volume 35 de *Lecture Notes in Electrical Engineering*, pages 201–219. 2009.

[14] Kaewtrakulpong P. et Bowden R.. An improved adaptive background mixture model for real-time tracking with shadow detection. Dans *Proceedings of the 2nd European Workshop on Advanced Video Based Surveillance Systems (AVBS '01)*, Kingston, UK, 2001.