Automatic Estimation of Multiple Motion Fields From Video Sequences Using a Region Matching Based Approach

Manya V. Afonso, *Member, IEEE*, Jacinto C. Nascimento, *Member, IEEE*, and Jorge S. Marques

***Abstract*—Estimation of velocity fields from a video sequence is an important step towards activity classification in a surveillance system. It has been recently shown that multiple motion fields estimated from trajectories are an efficient tool to describe the movement of objects, allowing an automatic classification of activities in the scene. However, the trajectory detection in noisy environments is difficult, usually requiring some sort manual editing to complete or correct them. This paper proposes two novel contributions. First, an automatic method for building pedestrian trajectories in far-field surveillance scenarios is presented not requiring user intervention. This basically comprises the detection of multiple moving objects in a video sequence through the detection of the active regions, followed by the estimation of the velocity fields that is accomplished by performing region matching of the above regions at consecutive time instants. This leads to a sequence of centroids and corresponding velocity vectors, describing the local motions presented in the image. A motion correspondence algorithm is then applied to group the centroids in a contiguous sequence of frames into trajectories corresponding to each moving object. The second contribution is a method for automatically finding the trajectories from a library of previously computed ones. Experiments on extensive video sequences from university campuses show that motion fields can be reliably estimated from these automatically detected trajectories, leading to a fully automatic procedure for the estimation of multiple motion fields.**

***Index Terms*—Region matching, trajectories, vector fields, video segmentation.**

I. Introduction

THE goal of a video surveillance system is to be able to detect and track moving entities (e.g. people, vehicles), to characterize typical behaviors and to detect abnormal situations, depending on the context. An automatic surveillance system should be able to learn typical behaviors from video data in an unsupervised manner, without involving specific knowledge about the actions performed by humans in the monitored environment. Such a system involves the following three steps: (a) segmentation of the video sequence to detect the objects of interest; (b) extraction of features (e.g. position, motion, shape); (c) features tracking; and (d) classification of the observed behavior based on the extracted features [38].

In outdoor applications, the object trajectories play an important role since they allow the system to characterize typical behaviors and discriminate abnormal ones. There are several ways to model trajectories in an image. For example, it was recently proposed the use of multiple motion fields, each of them representing a specific type of motion [25], for the static camera case. This model was applied with success to several problems. However, some of the training trajectories were hand edited to compensate for object detection and tracking errors.

Manuscript received January 03, 2013; revised April 30, 2013; accepted June 18, 2013. Date of publication September 06, 2013; date of current version December 12, 2013. This work was supported by FCT projects PEst-OE/EEI/LA0009/2013, and project 'ARGUS'-PTDC/EEA-CRO/098550/2008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Feng Wu.

The authors are with the Instituto de Sistemas e Robótica, Instituto Superior Técnico, 1049-001 Lisbon, Portugal (e-mail: mafonso@isr.ist.utl.pt; jan@isr.ist.utl.pt; jsm@isr.ist.utl.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2013.2281023

II. Related Work

We now describe most relevant published work in the field. Since our proposal combines different methods proposed in quite different contexts, we will describe the main contributions in those contexts that our proposal is enrolled, namely: segmentation, optical flow, region matching, and multiple motion fields.

A. Segmentation

Surveillance and monitoring systems require the segmentation of all moving objects in a video sequence. Indeed, segmentation is a key step since it influences the performance of the other modules. It aims to detect objects of interest in the video stream, using their visual and motion properties. It plays a key role since it reduces the amount of information to be processed by higher processing levels, e.g. object tracking, classification or recognition; and locates the position of the targets. A large spectrum of detection algorithms have been proposed, that can be broadly classified into the following classes: (i) *statistical approaches*, (ii) *non-statistical approaches*, and (iii) *spatio-temporal approaches*.

1) Statistical Based Approaches: In this class of works the background is modeled using a normal pdf e.g. [23], [36], [40], where each pixel is modeled as a Gaussian distribution [40] or a mixture of Gaussians [23], [36]. Also, minimization of Gaussian differences has been used [29]. Other type of statistical frameworks is the use of dynamic belief network dedicated to analyzing traffic scenes [18], or learning the chronological changes in the observed background scene in terms of distributions of multi-dimensional image vectors [31]. A combination of frame differences and statistical background models has also been proposed [9].

2) Non-Statistical Based Approaches: Some works use a deterministic background model e.g., by characterizing the admissible interval for each pixel of the background image as well

as the maximum rate of change in consecutive images or the median of largest inter-frames absolute difference [15], [16]. Background subtraction is a simple approach to detect moving objects in video sequences. The basic idea is to subtract the current frame from a background image and to classify each pixel as foreground or background by comparing the difference with a threshold [14]. Morphological operations followed by a connected component analysis are used to compute all active regions in the image. In practice, several difficulties arise: the background image is corrupted by noise due to camera movements and fluttering objects (e.g., trees waving), illumination changes, clouds, shadows, and other extraneous events. Another difficulty is the presence of *ghosts* [33], i.e., false active regions due to static objects belonging to the background image (e.g., cars) which suddenly start to move.

Several methods have been proposed, depending on the difficulty to be addressed. For instance, to deal with the ghosts several works are available, e.g. [8], [10], [11]. Concerning the shadows with non-stationary backgrounds, two types of changes have to be considered: slow changes (e.g., due to the sun motion) and rapid changes (e.g., due to clouds, rain or abrupt changes in static objects). Adaptive models and thresholds have been used to deal with slow background changes [4]. These techniques recursively update the background parameters and thresholds in order to track the evolution of the parameters in nonstationary operating conditions. To cope with abrupt changes, multiple model techniques have been proposed [4] as well as predictive stochastic models (e.g., AR, ARMA [24], [41]).

3) Spatio-Temporal Based Approaches: Another class of algorithms is based on spatio-temporal segmentation of the video signal. These methods try to detect moving regions taking into account not only the temporal evolution of the pixel intensities and color but also their spatial properties. Segmentation is performed in a 3D region of image-time space, considering the temporal evolution of neighbor pixels. This can be done in several ways e.g., by using spatio-temporal entropy, combined with morphological operations [22]. This approach leads to an improvement of the systems performance, compared with traditional frame difference methods. Other approaches are based on the 3D structure tensor defined from the pixels spatial and temporal derivatives, in a given time interval [35]. In this case, detection is based on the Mahalanobis distance, assuming a Gaussian distribution for the derivatives. This approach has been implemented in real time and tested with PETS 2005 data set. Other alternatives have also been considered e.g., the use of a region growing method in 3D space-time [37].

B. Optical Flow and Region Matching

Optical flow is a measure of the apparent motion of local regions of the image brightness pattern from one frame to the next. With some exceptions, it is an estimate of the motion field [3], [34]. Denoting the image intensity at position (x, y) and time t by $\mathbf{I}(x, y, t) \in \mathbb{R}$, the optical flow equation is

$$\nabla_x \mathbf{I}(x, y, t)u + \nabla_y \mathbf{I}(x, y, t)v + \nabla_t \mathbf{I}(x, y, t) = 0, \quad (1)$$

where (u, v) is the velocity vector at the position (x, y), and $\nabla_x \mathbf{I}$, $\nabla_y \mathbf{I}$, and $\nabla_t \mathbf{I}$ respectively denote the image gradient along the (x, y) and t directions. The differential operators ∇_x, ∇_y, and ∇_t assume that the image $\mathbf{I}$ is continuous on the x- and y-coordinates, and over time, but these operators can be approximated by differences or discrete filters in case $\mathbf{I}$ is a discrete image. This equation is not enough to obtain the velocity vector (u, v) associated to each image point since it has an infinite number of solutions, that is, the problem is ill-posed. This is known as the *aperture problem* [1]. Additional constraints have to be added (e.g., smoothness constraints).

There are several methods for computing the optical flow. Gradient based methods solve the optical flow (1), by imposing smoothness constraints on the field of velocity vectors. The Horn and Schunck method [17] uses a global smoothness constraint, requiring global optimization. The Lucas-Kanade method [21] assumes that the velocity is constant locally and combine local constraints over local regions. These methods are considered more accurate than the ones based on region matching, but work well for small displacements only [3].

There are also region-matching based methods which do not actually solve (1), but try to find the most likely position for an image region in the next frame. These methods are intuitively simple and relatively easy to implement.

Yet another method is the Bayesian multi-scale coarse to fine algorithm [34], that gets around the limitation of small displacements in the gradient based methods. A coarse to fine warping scheme involving two nested fixed point iterations for minimizing an energy functional that combines the assumptions of constant brightness and gradient, and a discontinuity-preserving spatio-temporal smoothness constraint was proposed in [6]. A variant of this method was proposed in [30], wherein video motion is represented using a set of particles and particle trajectories are optimized yielding the displacements as well as trajectories representing their motion.

The region matching is one of the main components of the proposed methodology. Region or template matching is an intuitively simple concept of locating a given region/template/object in an image. Logically, it can be used in a variety of problems—image registration, object detection, tracking, etc. In [7], a template matching based method was proposed for segmenting cell nuclei from microscopy images. Given an unlabelled image, template matching is used to determine which predetermined model for a nucleus best fits the new image, based on the Normalized Cross Correlation criterion. A hierarchical template matching based method was proposed in [20], to determine human activity by matching with a series of shapes extracted from known poses. The normalized histogram intersection was used as a similarity measure between active regions, used for region matching in a moving camera situation, in [13]. In [5], a method for region matching was proposed based on identifying the longest, best matching boundary parts of two regions in successive frames. Patch-wise self similarity measures for regions in images and video sequences, based on the sum of square differences (SSD) between patch colors were proposed as the criterion for matching in [32].

To our knowledge, related work closer to the method herein presented is proposed in [25], [26]. In these two works, the trajectories are also obtained via region matching, where the region association is achieved by a nearest neighbor mechanism. Here, we go further and able at building the trajectories in a robust fashion, since the region matching is based on a motion

correspondence (contrasting with nearest neighbor above mentioned). First, region matching between consecutive frames is achieved by minimizing a cost function. Then, motion correspondence is performed using the Veenman algorithm [39]. This will bring advantages over the previous work [25], [26] as stated in Section II-D.

C. Multiple Motion Field Model

Multiple motion fields were recently proposed as a tool to describe objects motion in a scene and to statistically characterize their trajectories [25]. The main characteristic of the method is that the object motion depends on the position in the image, i.e. each position $\mathbf{x}_t$ is assigned a displacement vector $\mathbf{T}_{k_t}$ that depends on $\mathbf{x}_t$. More specifically, this method takes in consideration a set of vector motion fields $\mathcal{T} = \{\mathbf{T}_1, \ldots, \mathbf{T}_K\}$, with $\mathbf{T}_k : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, for $k \in \{1, \ldots, K\}$. Denoting the velocity vector at position $\mathbf{x}_t$ by $\mathbf{T}_k(\mathbf{x})$, each object trajectory is generated according to the model

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{T}_{k_t}(\mathbf{x}_{t-1}) + \mathbf{w}_t, \tag{2}$$

where $k_t \in \{1, \ldots, K\}$ is the label of the active field at time t; $\mathbf{w}_t \sim \mathcal{N}(0, \sigma^2_{k_t}\mathbf{I})$ are independent samples of a zero-mean Gaussian random vector with covariance matrix $\sigma^2_{k_t}$. The label sequence $\mathbf{k} = (k_1, \ldots, k_n)$ is assumed to be a Markov chain of order one with space-dependent transition probabilities $\mathbf{B}_{ij}(\mathbf{x}_{t-1}) = P(k_t = j | k_{t-1} = i, \mathbf{x}_{t-1})$ where $\mathbf{B} : \mathbb{R}^2 \rightarrow \mathbb{R}^{K \times K}$ is a field of stochastic matrices. Thus, this model allows the switching probability to depend on the location of the object. The complete set of model parameters is $\boldsymbol{\theta} = (\mathcal{T}, \mathbf{B}, \boldsymbol{\sigma})$. These model parameters are defined in a $N \times N$ grid, and they have to be estimated at each node of the grid. Say that, the image is divided in a grid of $N \times N$ nodes. In each node the vector field is estimated, and in general, they are different since they depend on the current object position.

Since the label sequences of active fields, $\mathcal{K} = \{\mathbf{k}^{(1)}, \ldots, \mathbf{k}^{(S)}\}$ of the trajectories $\mathcal{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(S)}\}$ are unknown, the parameters are obtained using the EM algorithm [12]. Recall that, the label sequence is the sequence of the active fields in the samples of the trajectory. This is dependent of the object motion. Thus, each trajectory may contain different active fields through its samples that depends on the object displacements in the image.

The complete log-likelihood that characterizes the EM is $\mathcal{L} = \log p(\mathcal{X}, \mathcal{K}|\boldsymbol{\theta}) = \sum_{j=1}^{S} \log p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\boldsymbol{\theta})$ where each term $p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\boldsymbol{\theta})$ has the form[1]

$$p(\mathbf{x}, \mathbf{k}|\boldsymbol{\theta}) = p(\mathbf{x}_1)P(\mathbf{k}_1) \times \ldots \prod_{t=2}^{L_j} \underbrace{\mathcal{N}\Big(\mathbf{x}_t|\mathbf{x}_{t-1} + \mathbf{T}_{k_t}(\mathbf{x}_{t-1}), \sigma^2_{k_{t-1}}\mathbf{I}\Big)}_{\text{trajectory generation term (see 2)}} \times \underbrace{B_{k_{t-1},k_t}(\mathbf{x}_{t-1})}_{\text{switching term}}. \tag{3}$$

that contains the generation and switching of the trajectories. The E-step the expectation log-likelihood function (objective function) becomes

$$Q(\boldsymbol{\theta}; \widehat{\boldsymbol{\theta}}) \equiv \mathbb{E}\left[\log p(\mathcal{X}, \mathcal{K}|\boldsymbol{\theta})|\mathcal{X}, \widehat{\boldsymbol{\theta}}\right]$$

[1]dropping the superscript j for simplicity.

and in the the M-step, the model parameter estimates are updated according to

$$\widehat{\boldsymbol{\theta}}_{\text{new}} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \widehat{\boldsymbol{\theta}}) + \log p(\boldsymbol{\theta}). \tag{4}$$

where $\log p(\boldsymbol{\theta})$ is the prior. It is assumed that the motion fields are smooth, thus a Gaussian prior $\mathcal{N}(0, \alpha^2, \mathbf{I})$ is used, where α^2 is a global variance (i.e. regularization term) that weights the strength of the prior.

See [25], [27] how computation weights (E-step) as well as the maximization (M-step) is performed.

D. Contributions

Our contribution is to provide an efficient tool to automatically detect pedestrian trajectories from the video. Also, we show that these trajectories are a reliable input to further estimate the motion fields.

In this paper, we propose an automatic method for extracting object trajectories from the video sequence and compute the vector fields, without user intervention. We first detect the active regions by background subtraction, find the centroids of each 8-connected contiguous active region, and then compute the displacement at each centroid through region matching to estimate the motion fields. After computing the centroids of moving objects and their associated velocity fields for the entire sequence, we then perform a motion correspondence step to group matching points into trajectories.

The above methodology presents several advantages over existing work. For instance, in [25], [26], all the trajectories are collected first, then the estimation of the vector fields takes place. This happens, since the trajectories by simply performing a naive region association using a nearest neighborhood mechanism. This of course is prone to errors, requiring further manual intervention to correct these errors that this association may provide. Thus, detection failures may occur originating gaps within trajectories. This step (building the trajectories) is done off-line. After this task being accomplished, the estimation of the VF is performed.

Here, we surpass the difficulties above. Now, we are able to build the trajectories in a robust fashion, since we are using a region based algorithm based on a motion correspondence. First, the velocity fields are computed (see Section III-B) providing the best region matching between consecutive frames. Then, motion correspondence is performed using the Veenman algorithm [39]. The advantages of this procedure are the following:

1) We can perform the estimation of the VF in an on-line fashion (opposing to the off-line strategy). Since we do not find the need of any manual correction of the trajectories. Thus, we can estimate the VF as the trajectories are formed;
2) Also, we can avoid the gaps in the trajectories. If the active region is not detected at $t + 1$ time instant, we can assume the existence of an active region (with the size equal to the region in the previous step) and located in the vicinity given by the previous velocity city vector. The best location of the region at $\mathrm{t} + 1$ is achieved using (6), (7).

We also propose a method to automatically find the trajectory from the ground truth set that corresponds to each trajectory computed by our method, where the ground truth is the set of trajectories used in [25]–[27].

To testify the usefulness of the approach we provide an experimental methodology, in which (*i*) we validate the obtained trajectories (see Section V-A) and (*ii*) the resulting motion fields (see Section V-B).

The first and second tasks are accomplished as follows:

1) To accomplish (*i*) we perform a comparison between the trajectories used in [25]–[27], $\mathbf{X}_{\rm gt}$, taken as the ground truth and the obtained trajectories $\mathbf{X}_{\rm rm}$.[2] Here we compute the error statistics $E(\mathbf{X}_{\rm gt}, \mathbf{X}_{\rm rm})$ and we will show that indeed a small error is obtained (see Tables II, III),
2) For the second task in (*ii*) the following steps are accomplished:
   a) We first estimate the vector fields, say $\mathcal{T}_{\rm gt}$ using the EM algorithm as described in Section II-C with the ground truth trajectories $\mathbf{X}_{\rm gt}$,
   b) As above, we also estimate the vector fields $\mathcal{T}_{\rm rm}$ using the EM and taking the trajectories $\mathbf{X}_{\rm rm}$ (as the input) that are obtained with the propose approach.
   c) To ascertain the accuracy of the above vector fields $\mathcal{T}_{\rm gt}, \mathcal{T}_{\rm rm}$ for describing the trajectories, we plug in these fields, $(\mathcal{T}_{\rm gt}, \mathcal{T}_{\rm rm})$ into the predictive model (2), obtaining $\widehat{\mathbf{X}}_{\rm gt}$ and $\widehat{\mathbf{X}}_{\rm rm}$, respectively.
   d) Finally, we compute the signal to noise ratio $\mathrm{SNR}_{\rm gt}(\mathbf{X}_{\rm gt}, \widehat{\mathbf{X}}_{\rm gt})$ and $\mathrm{SNR}_{\rm rm}(\mathbf{X}_{\rm gt}, \widehat{\mathbf{X}}_{\rm rm})$ (see (20)).

From the aforementioned steps, it will be possible to certify how the methodology is accurate at estimating the trajectories. More specifically, obtaining similar values in $\mathrm{SNR}_{\rm gt}, \mathrm{SNR}_{\rm rm}$ means good performance. In this way, it is possible to show the reliability of the obtained trajectories that permit us to overcome the problems found in [25]–[27], where some trajectories required manual intervention.

E. Organization of the Paper

The paper is organized as follows. In the section above, we introduced preliminary concepts necessary for the completeness of the exposition of the proposed approach. In Section III, we present the details of the proposal. Section IV describes the way how we perform a comparison regarding the ground truth. Section V presents experimental results with real data concerning two different far-field scenarios and Section VI concludes the paper. Our approach assumes that the video frames are grayscale images.

III. Proposed Approach

In this section, we describe the segmentation and feature extraction processes. We first perform background subtraction to detect the active regions in a frame, and then estimate the velocity fields at the centroid of each active region using the region matching algorithm. The result of these steps is a sequence of vectors containing the spatial coordinates of the centroids of the active regions, over the entire set of frames, and also the corresponding velocity fields.

[2]the subscripts "gt" and "rm" stands for the "ground truth" and "region matching", respectively.

A. Active Region Detection

We represent a frame at time t with M rows and N columns by a matrix $\mathbf{I}_t \in \mathbb{R}^{M\times N}$. Given a sequence of T frames $\{\mathbf{I}_t, t = 1, \ldots, T\}$, we can estimate the background image $\mathbf{B} \in \mathbb{R}^{M\times N}$, if not known, by pixel-wise median filtering a sub-sampling of the frames. We segment each frame $\mathbf{I}_t$, by subtracting the background and thresholding the difference image with a predefined positive value λ, to produce a binary image $\mathbf{J}_t \in \{0,1\}^{M\times N}$ with the value at pixel m, n,

$$\mathbf{J}_t(m,n) = \begin{cases} 1, & \text{if } |\mathbf{I}_t(m,n) - \mathbf{B}(m,n)| > \lambda, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

For multiple moving objects, this requires finding the connected pixels above this threshold. We also do not assume beforehand, the number of moving objects. We therefore apply a clustering algorithm on this binary image $\mathbf{J}_t$ to find connected regions (assuming 8 neighbors), each cluster corresponding to a moving object.

In practice, because of the empirical nature of the threshold value λ, there may be false active regions (i.e. the threshold being exceeded where there is no motion) or the active region corresponding to a single moving object may get split into two or more clusters. We solve the second problem by dilating the binary image $\mathbf{J}_i$ before clustering, and solve the problem of false active regions by discarding the clusters with fewer than a certain threshold value of connected pixels. The structuring element used is a ball with a radius of 3 pixels, to have a smooth, rounded structure. We could also apply any algorithm that would make the binary image more piece-wise smooth. The number of active regions detected at the frame t, will be denoted by n_t. Fig. 6 illustrates some active regions or bounding boxes (in yellow rectangles, that are the current position of the pedestrians) and the corresponding trajectories (dots in the images) corresponding to past position of the pedestrians.

B. Region Matching

Region based matching approaches define the velocity vector of an object moving across successive frames as the vector of displacements (i, j) that produces the best fit between image regions at different times. The best fit means that a distance measure between a region in a frame, $\mathbf{I}_t$, and its possible location in the next frame, $\mathbf{I}_{t+1}$, is minimum for the vector (i, j), or a similarity measure such as cross-correlation is maximum. These methods are intuitively simple and relatively easy to implement, and the computational load is low since it has to be computed only for the active regions.

Let the bounding box of the k-th active region be $\mathcal{R}_k \in \mathbb{Z}^2$. Then assuming suitable boundary conditions (to handle the case when the template moves towards or close to the frame edges), the velocity vector is computed by solving the following optimization problem,

$$(u_k, v_k) = \arg \min_{i,j\in(-d_m, d_m)} E(i,j), \tag{6}$$

$$E(i,j) = \sum_{(m,n)\in\mathcal{R}_k} \left(I_t(m,n) - I_{t+1}(m+i, n+j)\right)^2 + \alpha\left(|i| + |j|\right). \tag{7}$$

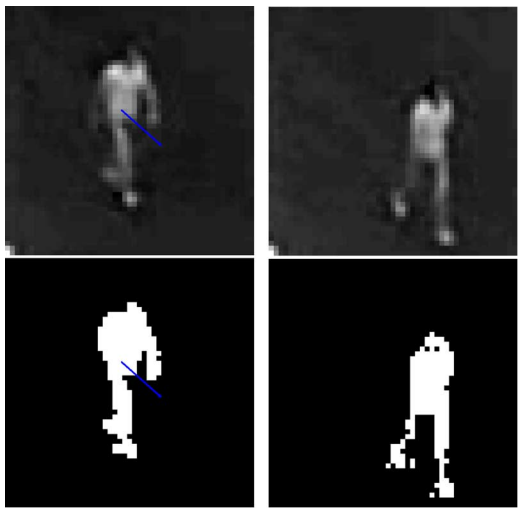


Fig. 1. Velocity field between 2 frames, for a man walking. (top left) grayscale frame at time t, (top right) grayscale frame at time $t+1$, (bottom left), (bottom right) active regions corresponding to the moving person in the two consecutive frames t and $t+1$.

where d_m is the maximum displacement, and $\alpha > 0$ is the regularization parameter that controls the relative weight of the penalty term, which in this case promotes sparsity of the solution; E is a vector containing the admissible displacements regarding the centroid of the bounding box (i.e. active region). Each value of E measures a possible displacement of the active region at $t+1$ time step. For each possible location (i, j) of the active region, we obtain a different value for the cost E. Thus, it is possible to obtain the best location (i.e. the best displacement of the active region)—the minimum of E.

Since the entire block is assumed to be displaced by the vector (u_k, v_k), we assume that this is the velocity at the centroid of the k-th region. It must be noted that the size of the block depends on the active region detected, and varies from frame to frame within a sequence.

Fig. 1 illustrates the velocity field between two frames, for a man walking with the velocity vector shown at the center of mass. The velocity vector is displayed as being concentrated at the centroid of the active region corresponding to the moving body. The figures on the left show the video frame and the binary image with the active region at time t, and those on the right show where the person has moved to at time $t+1$.

C. Motion Correspondence

At the end of the segmentation step and computation of the region matching, we have for a frame t, a set of n_t centroid locations $\{(x_i, y_i), i = 1, \ldots, n_t\}$ and their respective motion vectors $\{(u_i, v_i), i = 1, \ldots, n_k\}$. We therefore need to apply a region association algorithm to integrate the centroids in successive frames into trajectories. This differs from the approach presented in [28], where the pre-processing consists of two steps: active region detection using the Lehigh Omnidirectional Tracking System (LOTS) algorithm followed by region association. The approach herein proposed is threefold regarding the pre-processing in [28]: (1) it computes directly the vector fields, (2) it reduces drastically the errors that may arise in the region association mechanism, and (3) consequently avoids manual corrections to obtain the trajectories. In this paper, we apply the so called *Greedy Optimal Assignment* (GOA) *tracker* as proposed in [39] to perform the motion correspondence.

More specifically, let the vector $\mathbf{x}_k^t = [x_k, y_k]^T$ denote the position vector of the centroid of active region k in frame t. For the centroids of two regions i and j in two successive frames t and $t+1$, the association cost is

$$C_t(i,j) = \left\| \mathbf{x}_j^{t+1} - \mathbf{x}_i^t \right\|_2, \tag{8}$$

where $\|.\|_2$ is the ℓ_2-norm of a vector, which for a vector $\mathbf{x} = [x, y]^T \in \mathbb{R}^2$, is $\|\mathbf{x}\|_2 = \sqrt{x^2 + y^2}$.

This measure can be improved by taking into account the velocity vector $\mathbf{v}_k^t = [u_k, v_k]^T$ at the centroid of active region k in frame t, as follows [39]

$$C_t(i,j) = \left\| \mathbf{x}_j^{t+1} - \left(\mathbf{x}_i^t + \mathbf{v}_i^t\right) \right\|_2. \tag{9}$$

Let $\tilde{\mathbf{C}}_t$ be the size $n_{t+1} \times n_t$ matrix whose entries are computed using the above cost function. In general, the number of centroids in successive frames is not the same, that is, $n_{t+1} \neq n_t$. A column of this matrix corresponds to a centroid in frame t and a row corresponds to a centroid in frame $t+1$. A centroid in frame t may have a corresponding point in frame $t+1$ belonging to its trajectory or it may be the last point belonging to that particular trajectory. Likewise, a centroid in frame $t+1$ may belong to a trajectory having an associated point in frame t, or it could be the starting point of a new trajectory. To be able to account for the "birth" or "death" of trajectories, we pad the matrix $\tilde{\mathbf{C}}_t$ with entries equal to γ to form a square cost matrix $\mathbf{C}_t$ of size $(n_t + n_{t+1}) \times (n_t + n_{t+1})$,

$$\mathbf{C}_t = \left[\begin{array}{c|c} \tilde{\mathbf{C}}_t & \gamma \mathbf{1}_{(n_{t+1} \times n_{t+1})} \\ \hline \gamma \mathbf{1}_{(n_t \times n_t)} & \gamma \mathbf{1}_{(n_t \times n_{t+1})} \end{array} \right], \tag{10}$$

where $\mathbf{1}$ stands for a matrix whose entries are all 1, $\gamma > 0$ is the cost of starting a new trajectory or ending an existing one. The next step, is to determine for a given ith row which is the corresponding jth verifying a minimum entry condition, and thus, assigning the ith centroid in frame $t+1$ to an existing trajectory from frame t (if $j \leq n_t$), or to begin a new trajectory (if $j > n_t$). To accomplished this goal, the Hungarian matching algorithm [19] is used. More specifically, this is done by minimizing the assignment cost

$$\mathcal{C} = \sum_{i,j \in [1, n_t + n_{t+1}]} b_{ij} C_{ij} \tag{11}$$

under the constraint

$$b_{ij} \in \{0, 1\}, \quad \forall i, j, \tag{12}$$

$$\sum_{i=1}^{n_t + n_{t+1}} b_{ij} = 1, \quad \forall j, \tag{13}$$

$$\sum_{j=1}^{n_t + n_{t+1}} b_{ij} = 1, \quad \forall i. \tag{14}$$

IV. Matching With Ground Truth

This section describes the methodology used to perform a comparison between the generated trajectories as above described, and the ground truth trajectories.

Assuming the availability of a computed object's trajectory in a given video sequence (as described in Section III-C), we would like to compare it with the trajectory for the same object from the ground truth set. In the absence of such set, the ground truth is considered as the set containing the trajectories provided in [27], [28] with partial manual correction. This task should be automatically done without the user intervention, i.e. to manually locate the sequence of frames corresponding to the trajectory, in the video sequence. This has the possible problems that the trajectory and its counterpart from the ground truth may not exactly coincide in terms of starting and ending frames, may have some missed detections in some frames, or a single trajectory may have multiple trajectories in the ground truth corresponding to different segments (e.g. of an object moving in loops).

We denote an automatically detected trajectory by $\{\mathbf{x}_t\}_{t\in T}$, defined over a subset of frames $T = \{t_0, \ldots, t_M\}$ of the video sequence. Similarly, a trajectory i from the ground truth set is denoted as $\{\mathbf{x}_t^{gt,i}\}_{t\in T_i}$, with T_i being the set of relevant frames. If there are some frames in common between T and T_i, we define the set of overlapping frames, $T_{OL,i} = T \cap T_i$, and compute the mean square error (MSE) between the two trajectories at the times corresponding to the overlapping frames

$$E(\mathbf{x}^{gt,i}, \mathbf{x}_t) = \frac{1}{\#T_{OL,i}} \sum_{t\in T_{OL,i}} \left\| \mathbf{x}_t - \mathbf{x}_t^{gt,i} \right\|_2 . \quad (15)$$

If we have n ground truth trajectories which have some overlapping frames with trajectory $\{\mathbf{x}\}_t$, we compute this cost for each ground truth trajectory $i \in \{1, \ldots, n\}$, leading to the following n-length vector:

$$\mathbf{E}(\mathbf{x}_t) = \left[E(\mathbf{x}^{gt,1}, \mathbf{x}_t), E(\mathbf{x}^{gt,2}, \mathbf{x}_t), \ldots, E(\mathbf{x}^{gt,n}, \mathbf{x}_t)\right] \quad (16)$$

The matching trajectory from the ground truth set is the one that has the least MSE, that is,

$$\widehat{\mathbf{x}}_t = \arg \min_{i\in\{1,\ldots,n\}} \mathbf{E}(i). \quad (17)$$

with $\mathbf{E}(i) = E(\mathbf{x}^{gt,i}, \mathbf{x}_t)$ the ith element of $\mathbf{E}$.

During the matching procedure, three cases may happen in practice. This is illustrated in Fig. 2. In this figure the position of the object centroids is shown along the y-axis, with the x-axis corresponding to the frame number. The solid blue curve shows the evolution of an estimated trajectory over time, and the dashed red curve shows the match from the ground truth set. The three case are as follows:

- *case* #1: The simplest case is shown in a synthetic example in Fig. 2(a), i.e. when we have one estimated and one ground truth trajectories, eventually with different lengths. Here, the matching is straightforward, i.e. performed between the two above trajectories.
- *case* #2: When trajectory $\mathbf{x}_t$ is longer, it is possible that there are more than one ground truth trajectories that correspond to different segments of $\mathbf{x}_t$, which are disjoint in time (that is overlap with $\mathbf{x}_t$ over different sets of frames). This can happen, for instance, when the person or object moves in a loop or complete circuit which corresponds to two separate ground truth trajectories, for outbound and returning movement (see Fig. 5(b)). This situation is illustrated in Fig. 2(b). In this case, since the two ground truth trajectories correspond to non-overlapping segments of the estimated trajectory, we need to select both as matching trajectories.
- *case* #3: The process of matching is further complicated if the trajectories from the ground truth set have overlapping frames between them, as shown in Fig. 2(c). This is the general situation. In this case, we need to ensure that each point in $\mathbf{x}_t$ corresponds to another in at most one ground truth trajectory. Therefore, we first find all trajectories in the ground truth for which the MSE is within a limit of κ times the minimum value, $E_{\min}$, $\{i : \mathbf{E}(i) \leq \kappa E_{\min}\}$. The assignment of matching trajectories is then done as follows:

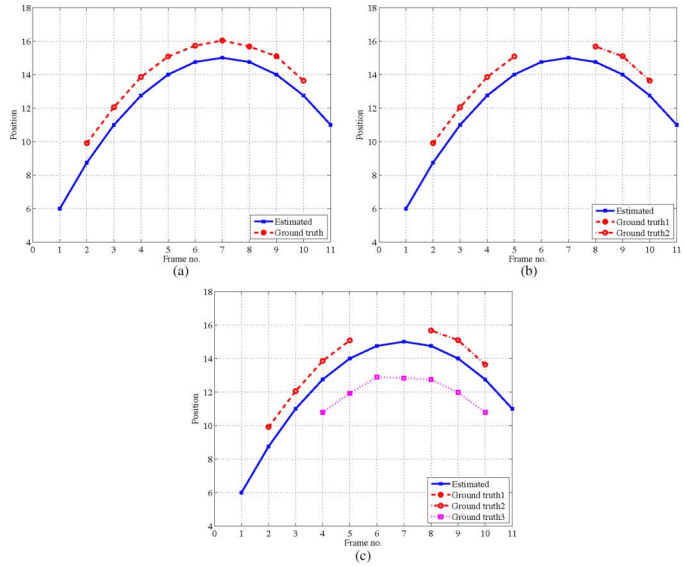


Fig. 2. Finding matching trajectories from the ground truth. (a) exactly one match, (b) multiple matches over non-overlapping segments, (c) overlaps between possible matches.

Algorithm *Matching with Ground Truth*

1. Assign ground truth set $\mathbf{X}_{gt} = \{i : \mathbf{E}(i) \leq \kappa E_{min}\}$.

2. **repeat**

3. Select the ith trajectory, $\mathbf{x}^{gt,i}$, from $\mathbf{X}_{gt}$ corresponding to the lowest MSE in $\mathbf{E}(i)$.

4. Assign trajectory $\mathbf{x}^{gt,i}$ as a match to the overlapping segment from the estimated trajectory $\mathbf{x}_t$.

5. Remove trajectory $\mathbf{x}^{gt,i}$ and all trajectories overlapping with it from $\mathbf{X}_{gt}$ and the corresponding elements of the vector $\mathbf{E}(i)$.

6. Remove corresponding MSE values from the vector $\mathbf{E}(i)$. If there is only one trajectory in the ground truth ($\mathbf{X}_{gt}$ is singleton), assign it to the overlapping segment from $\mathbf{x}_t$.

7. **until** the set $\mathbf{X}_{gt}$ is empty.

V. Experimental Evaluation

This section illustrates the effectiveness of the proposed approach on a set of video sequences of university campuses, Instituto Superior Técnico (IST), Lisbon and Universitat Politècnica de Catalunya (UPC), Barcelona.[3] Both sequences were recorded with a single static camera.

Several activities are considered depending on the scenario. Thus for the IST the following pedestrian's activities classes are (see Fig. 8(a)): (*i*) *crossing park down* (red) (*ii*) *passing through cars* (blue) (*iii*) *walking along* (cyan) (*iv*) *entering* (yellow) and (*v*) *leaving* (magenta). These are the most common trajectories performed by the pedestrians.

For the Barcelona scenarios the following activities are as follows (see Fig. 9(a)): (*i*) *up the steps* (cyan), (*ii*) *up the steps and turn right* (magenta), (*iii*) *walking along* (yellow) and (*iv*) *cross diagonally* (green).

Notice that, in both scenarios we are able to extract multiple trajectories and work with several trajectories at the same time. Fig. 6 shows several such situations. In Fig. 6(a) it is shown two pedestrians performing *walking along* (yellow) and *cross diagonally* type trajectories in the UPC scenario at the same time. Fig. 6(b) shows two pedestrians performing a *walking along* activity taken at different times.

Comparing the two scenarios, we see that (see Figs. 8(a), 9(a)) the trajectories in the UPC scenario exhibit higher dynamic perturbation than in the IST scenario. This means that the centroids of the bounding boxes in the UPC scenario fluctuate largely. The reason behind is that, when detecting the active regions, we may obtain partial detections, i.e. one bounding box that represents only the torso, or the legs. In this case, the association provides larger variations of the centroids in consecutive active regions when comparing with an ideal situation where the whole body is detected. This is illustrated in Fig. 6(c), (d). Notice that the detection of the pedestrian performing *walking along* activity varies significantly. In (c) we see yellow dots corresponding to the detection of the torso, and green dots corresponding to the detection of the legs. After a few frames, we can see that this trajectory is formed by the torso (yellow and cyan color) and legs (green color) detections. This makes the trajectory fluctuate largely.[4]

Another reason is that, we may have a group of two or five pedestrians. Here, the centroid can fluctuate between the pedestrians in the group, motivated by some partial detections. This situation is illustrated in Fig. 6(e). Here, the centroids may fluctuate within these pedestrians during a *walking along* trajectory. As such, the region association provides less smoothness in the trajectory. Since in the UPC scenario, the pedestrians are closer to the camera, than the IST scenario, these dynamical perturbations are more significant.

Additional information about these sequences is summarized in Table I. Because of the relatively long duration of the Barcelona sequence (9 hours), we estimate the background over each interval of 4000 frames, to account for changes in illumination conditions and shadows. Each background image is computed by median filtering over 40 frames. The value of the threshold λ in the segmentation step (5) was 12 for the IST sequence, and 50 for the UPC sequence. The minimum number of connected pixels to form an active region used was 40 for both sequences.

[3]The sequences at the UPC campus were acquired in the scope of the European Union project URUS, (FP6-EUIST-045062), http://urus.upc.es/.

[4]The colors used in this Fig. are not the same as shown in Figs. 3, 11, 10 for the sake of the explanation.

TABLE I
Information About the University Campus (IST and UPC) Sequences

Sequence	IST	UPC
Duration	46 minutes	9 hours
No. of frames	69,596	123,425
Frames per sec	25	4
Frame size	432×540	360×480
No. of activities	7	4

TABLE II
IST Sequence. Statistics of the Error in the Trajectories, With Respect to the Ground Truth

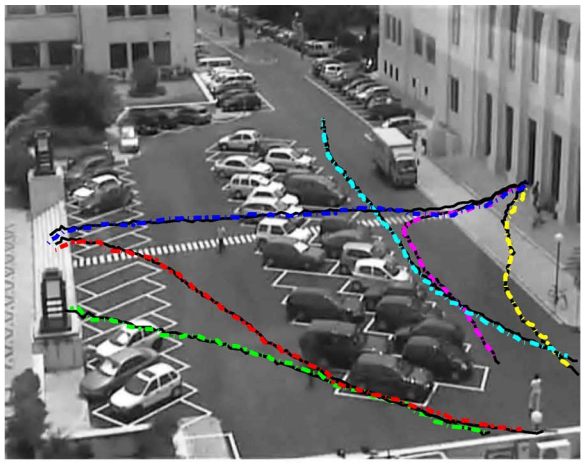
Trajectory class	Error Mean (pixels)	Error Standard Deviation (pixels)	Min error	Max error
entering	4.03	0.99	2.49	10.47
leaving	3.80	1.07	2.94	4.31
walking along	2.46	0.35	1.89	2.67
crossing park up	4.12	1.29	2.86	5.11
crossing park down	5.33	1.80	2.90	9.37
passing through cars	4.49	0.87	3.15	4.77
browsing	3.23	0.82	2.05	3.90
Overall	4.31	1.59		

A. Validation of Trajectories

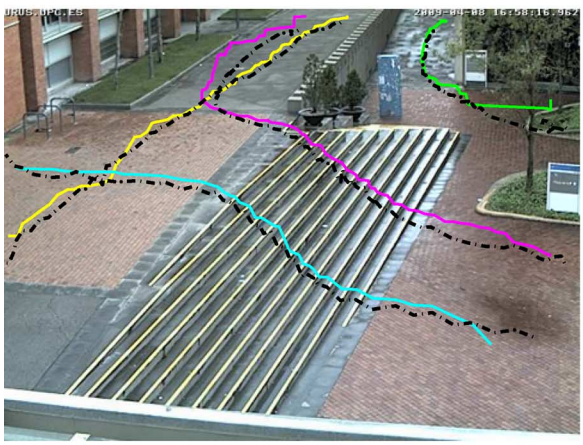
In this section, we compare the accuracy of the trajectories obtained using our method by computing the error statistics with respect to the ground truth of the trajectories. We compute the mean and standard deviation of the error, averaged over all trajectories which correspond to ground truth trajectories belonging to each of the classes of activities above described. These values, the overall error mean and standard deviation (over all the trajectories), and the best and worst case error values for the IST campus scenario are presented in Table II.

Fig. 3 shows examples of trajectories from the ground truth set (indicated by black solid curves) and their corresponding trajectories obtained by the proposed method (colored dashed curves) for both scenarios. It can be seen a trustfully matching is obtained for the estimated trajectories.

Fig. 4 shows two trajectories from the IST sequence, corresponding to people walking, detected over a segment of 2000 frames, and their respective velocity vectors superimposed. The velocities were scaled by a factor of 10 and subsampled by a factor of eight for the purpose of display, so that the arrows would be noticeable but not to crowd the display with too many arrows. The trajectory in Fig. 5(a) was found by the library search method corresponding to a single trajectory from the ground truth library, with a root mean square error (RMSE) of 4.53 pixels. The other trajectory of a person leaving the building and entering it again, which appears as a loop in Fig. 5(b) was found to have two non-overlapping matches in the ground truth (corresponding to activities "entering" and "leaving"), which are 70 frames apart. Therefore, for the estimation of the motion fields using the EM algorithm, we split this trajectory into two segments corresponding to the two ground truth trajectories, "leaving" with an RMSE of 3.99, and "entering", with an RMSE of 2.97.



(a)



(b)

Fig. 3. Examples of matched trajectories for each class of activities. The solid black curves indicate ground truth trajectories and the colored dashed curves indicate the trajectories estimated using the proposed approach. (a) IST Campus. Colors indicate: entering: yellow, leaving: magenta, walking along: cyan, crossing park up: green, crossing park down: red, passing through cars: blue; (b) UPC Barcelona. Colors indicate:up the steps and turn right: magenta, walking along: yellow, up the steps: cyan, upper cross diagonally: green.

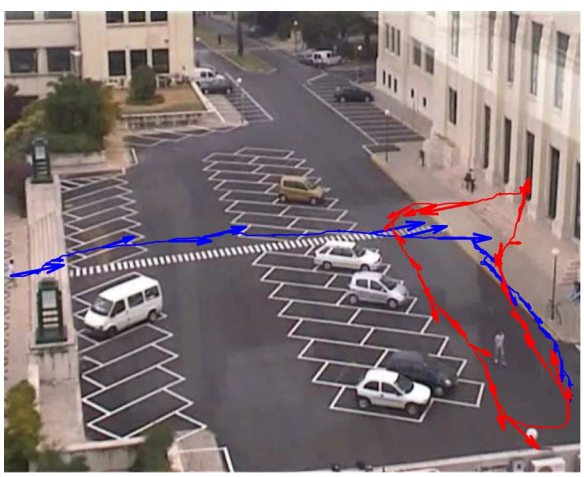
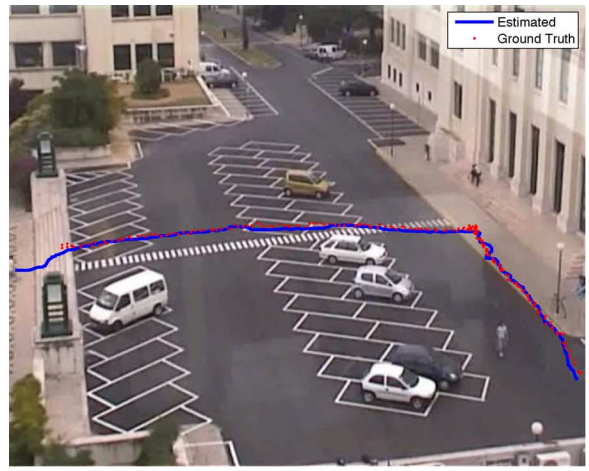
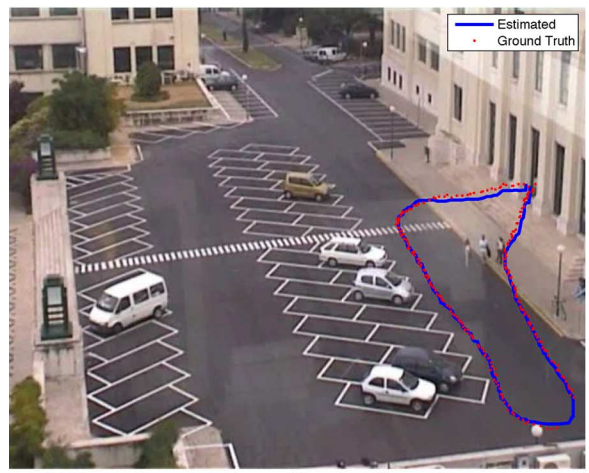


Fig. 4. Motion fields computed using region matching superimposed over the corresponding trajectories (scaled by a factor of 10 for display).

Figs. 8 and 9 show all the trajectories from the ground truth set and set of trajectories obtained with the proposed method, for the IST campus and Barcelona sequences, respectively. The



(a)



(b)

Fig. 5. Extracted trajectories using the proposed approach (blue solid curves) and their corresponding trajectories from the ground truth (red dotted curves).

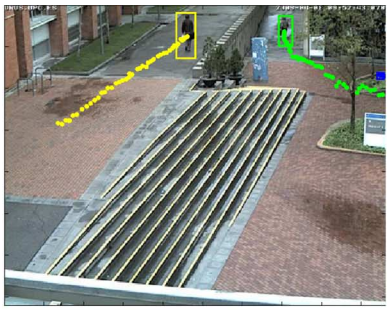
top rows show the trajectories superimposed on the starting frame as captured by the camera, and the bottom rows show the superimposed trajectories after applying the homography (see Section V-B). The left and right columns of the Figs. 8, 9 show the ground truth and the estimates trajectory sets, respectively. Each color corresponds to a class of activities.

Tables II, III show the error statistics for both scenarios. The mean and standard deviation are averaged over all trajectories. It is seen that the method presented herein allows at obtaining remarkable accuracy. These results (results are in pixels) should be compared taking in consideration the image size of the scenario (see Table I). For instance, in the IST scenario a small error of 2–5 pixels is obtained for the 370 trajectories of the IST scenario.

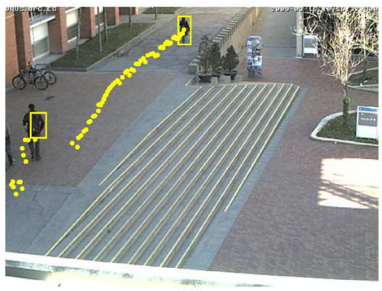
B. Validation of Motion Fields

In this section we aim to validate the motion fields obtained with the proposed approach.

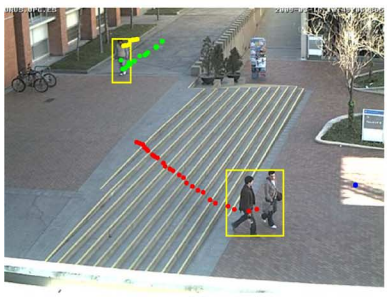
Before estimating the vector fields, we apply a projective transformation (homography) between the image and a plane parallel to the ground. This is done to achieve viewpoint invariance, by projecting all image measurements onto a view orthogonal to the ground plane. We find this pre-processing mandatory, since in the absence of the homography, the performance



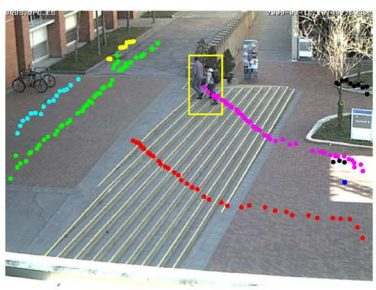
(a)



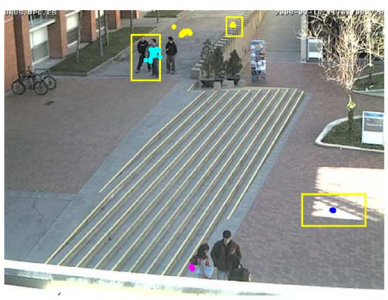
(b)



(c)



(d)



(e)

Fig. 6. Several situations shown during the extraction of multiple trajectories: (a) *walking along* (yellow) and *cross diagonally* (green) activities, (b) two simultaneous *walking along* activities, (c,d) perturbations in the *walking along* trajectory, where the detection occur at different body parts (also other trajectories occur at the same time), (e) fluctuations of the centroid in a group of pedestrians (also note for some false positives detected).

TABLE III
UPC BARCELONA SEQUENCE. STATISTICS OF THE ERROR IN THE TRAJECTORIES, WITH RESPECT TO THE GROUND TRUTH

Trajectory class	Error Mean (pixels)	Error Std Dev. (pixels)	Min error	Max error
up the steps and turn right	2.21	0.19	1.04	4.89
walking along	2.00	0.69	0.35	6.83
up the steps	2.77	0.75	0.62	7.78
cross diagonally	1.97	0.39	0.39	6.09
Overall	2.16	0.64		

of theEM decreases when estimating the vector fields. This happens when the pedestrian(s) are far from the camera view-point. Under these circumstances, the velocity is almost null. This hampers the accuracy in estimating the velocity fields. With the homography we can circumvent this difficulty.

Basically, this works as follows. To achieve viewpoint invariance, all image measurements are projected onto a view orthogonal to the ground plane (*bird's eye* view), using a projective transformation (homography) between the image and a plane parallel to the ground. The parameters of this projection were obtained by considering a set of points in the scene with known ground-plane coordinates. The homography is defined as follows

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}, \qquad (18)$$

where $[X\ Y]^T$ and $[x\ y]^T$ are the coordinates in the real world and in the image plane respectively. Since the non singular homogeneous matrix H has 8 degrees of freedom, four points are needed to determine them uniquely. Figs. 7 illustrates the homography of the two scenarios used in the experimental evaluation.

For the validation of the vector fields $\mathcal{T}_{\rm gt}$ we use the EM algorithm as described in Section II-C using the homography of the ground truth $\mathbf{X}_{\rm gt}$ trajectories. Using the same procedure, we also estimate the vector fields $\mathcal{T}_{\rm rm}$ using the obtained region matching trajectories $\mathbf{X}_{\rm rm}$. We then plug in the obtained fields $\mathcal{T}_{\rm gt}, \mathcal{T}_{\rm rm}$ into (2) allowing to obtain the sets of trajectory estimates, $\widehat{\mathbf{X}}_{\rm gt}$ and $\widehat{\mathbf{X}}_{\rm rm}$, respectively. We then compute the $\mathrm{SNR}(\mathbf{X}_{\rm gt}, \widehat{\mathbf{X}}_{\rm gt})$ and $\mathrm{SNR}(\mathbf{X}_{\rm gt}, \widehat{\mathbf{X}}_{\rm rm})$.

Notice that, when estimating the vector fields we do not know beforehand, what is most suitable number of models (i.e. model order) to estimate them. This happens since, we have a training set in which the samples from different class of trajectories are mixed together (i.e., some class-trajectories may require higher number of models than the others). In a classification context, the common strategy [25], [26] is to vary the model order in a pre-defined range $k \in \{1, \ldots, K\}$ and estimate the model parameters for each value of k. Then, it is performed a classification task in a disjoint test set also varying k in the above range. Finally, the value of k that maximizes the classification is chosen. This is a discriminative model selection for density models which aims at choosing the model order that achieves the best recognition accuracy. For this reason we also adopt a similar strategy (i.e. we vary the model order) since the performance at validating the vector fields depends on the value of k.

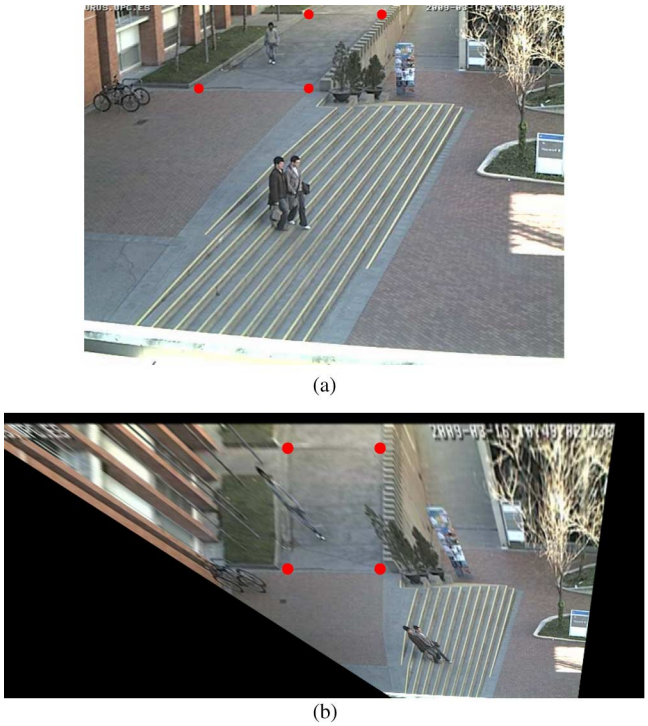


Fig. 7. Two images showing the UPC scenario before (a) and after (b) the projective transformation. The four points (shown in red dots) in the real world (a) and in the image plan (b) contain the coordinates to perform the transformation.

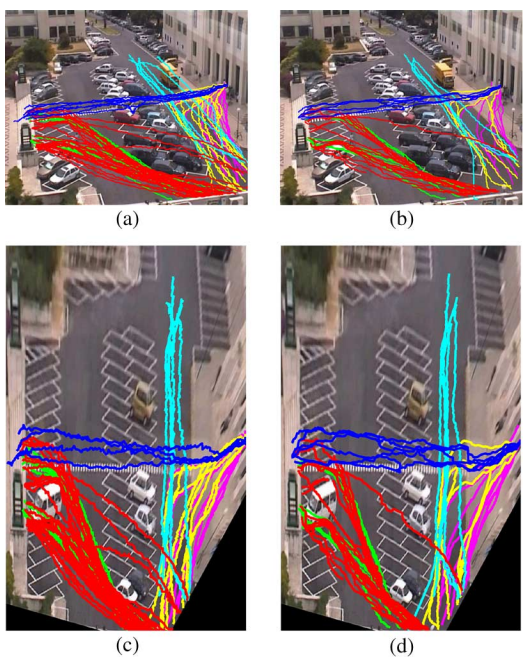


Fig. 8. IST campus scenario. Trajectories from (a),(c) the ground truth and (b),(d) extracted set, superimposed on the starting frame, before (top) and after (bottom) applying a homographic transformation. Colors indicate: entering: magenta, leaving: yellow, walking along: cyan, crossing park up: green, crossing park down: red, passing through cars: blue.

Also note that, it is not possible to directly compare the vector fields obtained from both sets of trajectories, since the field estimates depend on the initialization of the EM method. We therefore, evaluate each set of vector fields by measuring its ability

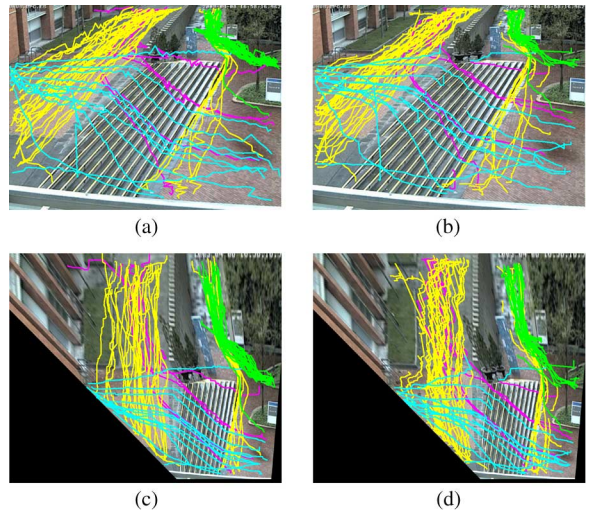


Fig. 9. UPC, Barcelona campus scenario. Trajectories from (a),(c) the ground truth and (b),(d) extracted set, superimposed on the starting frame, before (top) and after (bottom) applying a homographic transformation. Colors indicate: up the steps and turn right: magenta, lower left to upper right: yellow, up the steps: cyan, upper right to top: green.

to predict the target position at the next time instant. According to the dynamic model in (2) the prediction error can be written as

$$\hat{\mathbf{e}}_t = \mathbf{x}_t - \mathbf{x}_{t-1} - \mathbf{T}_{k_t}(\mathbf{x}_{t-1}). \tag{19}$$

In practice, we do not know which field $k_t \in \{1, \ldots, K\}$ is active at time instant t. Therefore, we select the error with the smallest norm (ideal switching). We can now define an SNR measure given by

$$\mathrm{SNR}(\mathbf{X}_{\mathrm{gt}}, \widehat{\mathbf{X}}_{\mathbf{m}}) = 10 \log_{10} \left(\frac{\sum_{t=2}^{L} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2}{\sum_{t=2}^{L} \min_k \left\| \mathbf{x}_t - \mathbf{x}_{t-1} - \mathbf{T}_{k_t}^{(\mathbf{m})}(\mathbf{x}_{t-1}) \right\|^2} \right). \tag{20}$$

where $\mathbf{m}$ can be either "gt" or "rm" depending on which trajectories are used (i.e. ground truth or region matching). The same number of trajectories per activity was used to train the model for both the ground truth trajectories and those extracted using the proposed approach.

For illustration purposes, Fig. 10 shows the motion fields estimated using the EM algorithm, trained with trajectories corresponding to a single activity from the IST set, using a single model. Fig. 10(a) and 10(b) show the motion fields for the activity "entering" using the trajectories from the ground truth set, and using the trajectories obtained using the proposed approach, respectively. Similarly, Figs. 10(c) and 10(d) present the corresponding motion fields for the activity "passing through cars". Two activities from the Barcelona set and their corresponding motion fields are similarly presented in Fig. 11.

One of the goals of this work is to evaluate how well the motion fields obtained from the trajectories in the training set describe the trajectories from the test set. To obtain the SNR measure in (20), we need to ensure that the trajectories used for training are not present in the test set. We therefore use five fold

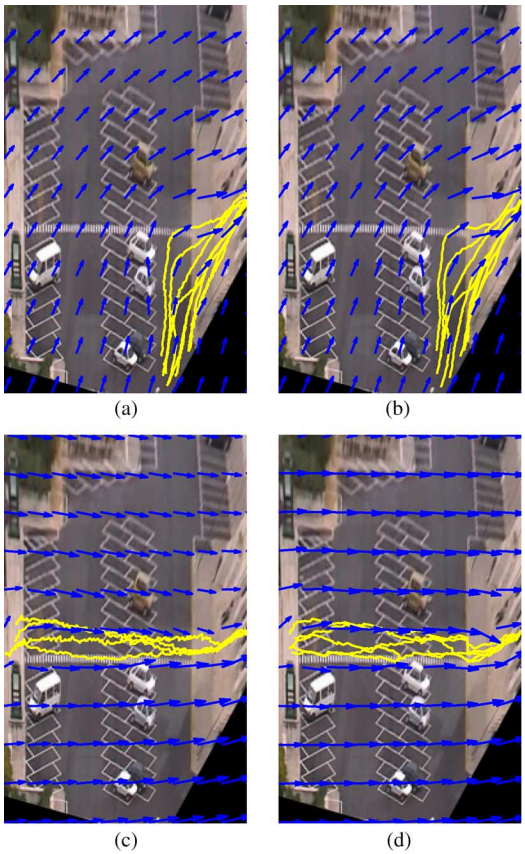


Fig. 10. IST campus scenario. Motion fields for trajectories corresponding to 2 different activities. "Entering": (a) estimated from the GT trajectories, (b) estimated from trajectories extracted using the proposed approach. "Passing through cars": (c) estimated from the GT trajectories, (d) estimated from trajectories extracted using the proposed approach.

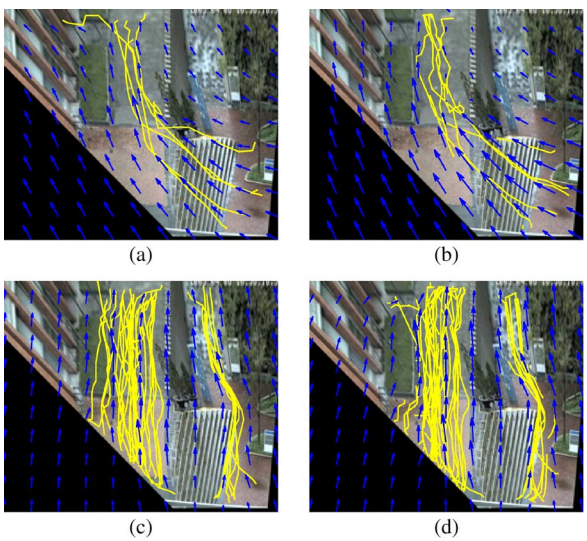


Fig. 11. UPC Barcelona scenario. Motion fields for trajectories corresponding to 2 different activities. "up the steps and turn right": (a) estimated from the GT trajectories, (b) estimated from trajectories extracted using the proposed approach. "lower left to upper right": (c) estimated from the GT trajectories, (d) estimated from trajectories extracted using the proposed approach.

cross validation strategy, in which the set of trajectories (ground truth or estimated) are split into five folds (one for test and the remaining for training). Also, the SNR (see (20)) depends on the number of models. So we define the range for $k \in \{1, \ldots, 8\}$.

TABLE IV
IST SEQUENCE. SNR OBTAINED IN 5-FOLD CROSS VALIDATION, FOR DIFFERENT NUMBERS OF MODELS WITH THE DISTANCE PARAMETER SET TO $\kappa E_{\min} = 15$

number of models	2	3	4	5	6	7	8
GT	3.64	3.4	5.41	3.32	3.3	2.91	3.33
	4.1	5.78	6.41	3.74	3.74	3.7	3.6
	3.98	3.81	6.48	3.55	3.75	3.71	4.96
	4.48	4.35	6.67	3.94	3.89	4.5	3.92
	4.29	4.18	6.25	4.07	4	3.92	4.77
GT avg.	**4.10**	**4.3**	**6.24**	3.72	3.74	**3.75**	**4.12**
Region Matching	3.68	3.34	5.95	5.35	3.27	3.07	2.85
	4.17	4.95	5.73	3.57	3.65	3.61	2.27
	3.85	3.64	5.76	3.3	3.65	3.28	3.59
	4.41	4.26	6.3	3.9	3.86	3.82	3.78
	4.41	4.26	5.43	3.93	5.58	3.57	4.11
RM avg.	**4.10**	4.09	5.83	**4.01**	**4.00**	3.47	3.32

TABLE V
IST SEQUENCE. SNR OBTAINED IN 5-FOLD CROSS VALIDATION, FOR DIFFERENT NUMBERS OF MODELS WITH THE DISTANCE PARAMETER SET TO $\kappa E_{\min} = 20$

number of models	2	3	4	5	6	7	8
GT	3.61	3.4	5.4	3.31	3.29	3.13	5.24
	4.1	5.82	6.34	3.74	3.7	3.66	3.61
	3.98	3.81	6.47	3.78	3.75	3.42	4.34
	4.46	5.97	6.63	3.97	3.93	3.88	3.43
	4.57	4.43	6.45	5.95	4.17	4.18	3.95
GT avg.	4.15	**4.68**	**6.26**	4.15	**3.77**	3.65	**4.11**
RM	4.01	3.67	5.02	3.39	3.19	4.75	2.69
	4.29	4.01	5.65	6.03	3.42	3.22	4.56
	3.87	3.67	5.75	3.33	3.2	5.1	3.67
	4.46	4.32	6.11	4.15	3.8	3.67	3.24
	4.61	4.45	5.68	4.3	3.66	6.53	3.81
RM avg.	**4.25**	4.02	5.64	**4.24**	3.46	**4.65**	3.6

In this procedure, the data partition is balanced, such that, in each training round all the trajectories classes are considered.

For fairness of training data size, we discard the ground truth trajectories which do not have a match from the set estimated using our method, found using the library search method described in [2]. Hence, both our data sets have N trajectories, $\mathbf{X}_{\mathrm{rm}} = \{\mathbf{x}_{\mathrm{rm},1}, \ldots, \mathbf{x}_{\mathrm{rm},N}\}$, and $\mathbf{X}_{\mathrm{gt}} = \{\mathbf{x}_{\mathrm{gt},1}, \ldots, \mathbf{x}_{\mathrm{gt},N}\}$.

Tables IV and V present the SNR values obtained for each fold and the average SNR value (over the five folds), for the IST sequence. We varied the distance threshold parameter $\kappa E_{\min}$ from algorithm *Matching with Ground Truth* in the interval $\kappa E_{\min} = \{15, \ldots, 20\}$.[5] The Tables IV and V show the obtained results for the two limit values of this interval. We see that the average value of the SNR (over the 5 folds) obtained with the proposed method was comparable to that achieved with the ground truth, and that there was no significant change in using a tighter criterion for matching trajectories between the ground truth and estimated set. This happens since the distance between the pedestrians and the camera is significantly large. Tables VI and VII present the corresponding results for the UPC Barcelona data set with parameter $\kappa E_{\min} \in \{8, \ldots, 15\}$, respectively. Here too, the average SNR obtained with the proposed method was comparable to that achieved with the ground truth. Also, we observe that the performance has a larger variation (regarding the IST scenario) with $\kappa E_{\min}$, since

[5]The range of the parameter depends on the relative distance of the pedestrian to the camera, i.e. the size of the foreground region regarding the size of the image.

TABLE VI
UPC BARCELONA SEQUENCE. SNR OBTAINED IN 5-FOLD CROSS VALIDATION, FOR DIFFERENT NUMBERS OF MODELS WITH THE DISTANCE PARAMETER SET TO $\kappa E_{\min} = 8$

number of models	2	3	4	5	6	7
GT	3.74	5.03	5.77	6.59	6.15	6.5
	3.83	4.93	6.17	7.1	6.87	6.33
	3.86	4.78	5.85	5.95	6.03	6.8
	3.8	4.47	6.17	5.84	6.77	7.05
	4.26	4.9	5.81	6.62	5.86	6.69
GT avg	**3.89**	**4.82**	**5.95**	6.42	6.34	6.67
RM	2.59	3.8	4.76	6.9	6.96	7.11
	2.62	4.52	4.42	6.86	7.31	7.03
	2.72	4.78	5.44	7.08	7.06	7.41
	2.66	4.61	4.81	6.93	7.05	7.28
	2.64	4.36	4.73	6.7	6.74	6.89
RM avg	2.65	4.41	4.83	**6.89**	**7.02**	**7.14**

TABLE VII
UPC BARCELONA SEQUENCE. SNR OBTAINED IN 5-FOLD CROSS VALIDATION, FOR DIFFERENT NUMBERS OF MODELS WITH THE DISTANCE PARAMETER SET TO $\kappa E_{\min} = 15$

number of models	2	3	4	5	6	7
GT	7.62	8.05	9.11	10.05	9.51	9.55
	6.23	7.1	7.72	8.84	8.89	8.52
	6.52	7.32	8.33	8.81	7.67	8.91
	6.21	6.98	6.55	7.1	7.41	9.05
	7.33	8.47	9.01	10.1	10.67	11.2
GT avg.	**6.78**	**7.58**	**8.14**	8.98	8.83	9.45
RM	7.57	8.08	8.27	9.62	9.85	10.78
	6.25	6.37	7.18	9.14	8.61	9.48
	6.62	8.06	7.92	8.59	9.03	9.21
	6.27	6.58	7.16	8.59	8.44	8.47
	7.17	7.8	8.67	9.65	10.02	10.28
RM avg.	**6.78**	7.38	7.84	**9.12**	**9.19**	**9.64**

now the pedestrians are placed closer to the network camera. In both of the scenarios, we conclude that the proposed framework exhibits a good SNR regarding the trajectories of the ground truth, testifying its truthfulness at the trajectory recovering.

As a final remark we should highlight that the methodology herein proposed is in accordance with the ground truth data. Notice that the SNR using the ground truth and the SNR using the proposed region matching criterion, reach the best scores for the same value of k. In the Campus scenario we obtained the best score for $k = 4$, and for the UPC scenario we obtained the best SNR for $k = 7$.

C. Activity Classification

In this section we present additional experiments which aim to illustrate activity (trajectory) classification. We report classification results using the ground truth trajectories and the trajectories obtained with the proposed region matching framework. More specifically, we compare the results using the method in [25], in which non-parametric vector fields are used to describe the trajectories. We adopt the UPC scenario where, at the same time we hope to achieve similar accuracy for trajectory classification.

This example is challenging due to the similarity of the motion between trajectory classes. For instance, the motion between the activities *up the stairs* and *up the stairs and turn right* are similar. Both contain a "left" motion (when stepping the stairs). Similarly, for the classes *walking along* and *up the stairs and turn right* where the "up" motion is present in both. This of course, makes difficult the classification task.

TABLE VIII
ACCURACY (IN %) FOR THE GROUND-TRUTH (GT) TRAJECTORIES (TOP) AND REGION MATCHING (RM) TRAJECTORIES (BOTTOM)

N^o of models	3	4	5	6
GT	73.9%	78.6%	88.3%	78.3%
RM	71.8%	75.6%	87.8%	74.4%

TABLE IX
CLASSIFICATION RESULTS USING NON-PARAMETRIC MODELS USING [25] WITH THE GROUND TRUTH TRAJECTORIES (TOP TABLE) AND WITH THE PROPOSED REGION MATCHING METHOD (BOTTOM TABLE). FOUR ACTIVITIES ARE CONSIDERED AT THE UPC SCENARIO: $a_1 \rightarrow$ *walking along*, $a_2 \rightarrow$ *up the steps*, $a_3 \rightarrow$ *up the steps and turn right*, $a_4 \rightarrow$ *cross diagonally*

	Classification with ground-truth			
	a_1	a_2	a_3	a_4
a_1	**100**%	0%	0%	0%
a_2	6.06%	**69.7**%	0 %	24.24%
a_3	16.67%	0%	**83.33**%	0%
a_4	0%	0%	0%	**100**%
	Classification with region-matching			
	a_1	a_2	a_3	a_4
a_1	**85.71**%	0%	0%	14.29%
a_2	9.09%	**81.82**%	2.27 %	6.82%
a_3	5.88%	0%	**88.24**%	5.88%
a_4	0%	4.55%	0%	**95.45**%

As in [25], the procedure to perform the classification follows the same strategy.[6] We used a five fold cross validation strategy to obtain the performance accuracy. For each of the five rounds, a training and a testing sets are used. The data partition is balanced, such that, in each training round all the trajectories classes are considered. In the training stage, the model parameters that characterizes the multiple vector fields are estimated. The model includes (*i*) the motion fields, (*ii*) noise standard deviations and (*iii*) spacevarying switching matrices. Besides this set of parameters, we also have to specify the number of motion models k. To accomplish this, the underlying assumption is that, we make use of the knowledge that the obtained model is going to be used for a specific task, in this case, a classification task. The goal, is thus, to select the generative model that achieves the best classification accuracy. In practice we varied $k \in \{1, \ldots, 6\}$. For $k \in \{1, 2\}$ we obtained accuracy of 20% for both of the methodologies. This means that the number of models do not suffice to describe all motion regimes presented in the four trajectory classes. Thus, we do not detail the results for these two values. Table VIII shows a comparison between the two methods for $k \in \{3, \ldots, 6\}$. Although, the classification accuracy using the proposed region matching trajectories is lower for all values of k, we can say that indeed the obtained performance is competitive regarding the ground truth trajectories. Notice that once more, the two methodologies are compatible when choosing the best model order to describe the vector fields, i.e. $k = 5$.[7] Table IX details the performance for each

[6]We refer to the reader [25] for an in deep review concerning the trajectory classification methodology.

[7]The values of the Table VIII correspond to the mean value of the diagonal entries of the confusion matrix.

TABLE X
COMPUTATION TIMES FOR THE VARIOUS STEPS (IN SECONDS)

Seq.	Region Matching	RM per frame	Motion Corr.	Library Search	EM (nM) 1	2	3	4	5	6	7
IST Campus	576.49 min	2.49	325.16	2.29	22.25	48.53	66.01	84.18	102.52	180.82	208.62
Barcelona	725.90 min	0.35	1116.35	6.37	7.58	16.20	22.31	28.20	34.53	57.74	66.08

activity in the scenario for the best model order (third column of the Table VIII).

We present the computation times for each of the steps—region matching, motion correspondence, library search, and the EM algorithm (with the number of models varying from 1 to 7), in Table X.

VI. CONCLUSIONS

We have proposed a method for automatically computing the trajectories and velocity fields of multiple moving objects in a video sequence, using region matching. The trajectories obtained were found to be close to the manually edited ground truth trajectories, for a large set of activities occurring in the video sequences. The motion fields estimated from the automatic trajectories using the EM method were found to lead to an SNR close to that obtained with the ground truth trajectories. Also, we conclude that for the trajectory classification the proposal achieves comparable results. This suggests that the proposed methodology is reliable for a fully automatic extraction of multiple motion fields. This is encouraging for the extension of this framework to denser environments such as crowds of moving people.

REFERENCES

[1] E. Adelson and J. Movshon, "Phenomenal coherence of moving visual patterns," *Nature*, vol. 300, no. 5892, pp. 523–525, 1982.

[2] M. V. Afonso, J. S. Marques, and J. C. Nascimento, "Automatic estimation of multiple motion fields using object trajectories and optical flow," in *Proc. Int. Conf. Pattern Recognition Applications and Methods (ICPRAM 2012)*, Vilamoura, Portugal, 2012.

[3] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput. Vision*, vol. 12, pp. 43–77, 1994.

[4] T. Boult, R. Micheals, X. Gao, and M. Eckmann, "Into the woods: Visual surveillance of non-cooperative camouflaged targets in complex outdoor settings," *Proc. IEEE*, vol. 89, no. 10, pp. 1382–1402, Oct. 2001.

[5] W. Brendel and S. Todorovic, "Video object segmentation by tracking regions," in *Proc. IEEE 12th Int. Conf. Computer Vision, 2009*, 2009, pp. 833–840.

[6] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. ECCV (4)*, 2004, pp. 25–36.

[7] C. Chen, W. Wang, J. A. Ozolek, and G. K. Rohde, "A flexible and robust approach for segmenting cell nuclei from 2d microscopy images using supervised learning and template matching," *Cytometry Part A*, 2013.

[8] Collins, Lipton, Kanade, Fujiyoshi, Duggins, Tsin, Tolliver, Enomoto, and Hasegawa, A System for Video Surveillance and Monitoring: Vsam Final Report Robotics Institute, Carnegie Mellon University, Tech. rep. CMU-RI-TR-00-12, May 2000.

[9] R. Collins, A. Lipton, and T. Kanade, "A system for video surveillance and monitoring," in *Proc. American Nuclear Society (ANS) 8th Int. Topical Meeting Robotic and Remote Systems*, Pittsburgh, PA, USA, Apr. 1999, pp. 25–29.

[10] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts and shadows in video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, 2003.

[11] R. Cucchiara, C. Grana, and A. Prati, "Detecting moving objects and their shadows: An evaluation with the PETS2002 dataset," in *Proc. 3rd IEEE Int. Workshop Performance Evaluation of Tracking and Surveillance (PETS 2002) in Conj. With ECCV 2002*, Pittsburgh, PA, USA, May 2002, pp. 18–25.

[12] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. Roy. Statist. Soc., Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[13] J. Fauqueur, G. Brostow, and R. Cipolla, "Assisted video object labeling by joint tracking of regions and keypoints," in *Proc. IEEE 11th Int. Conf. Computer Vision, 2007 (ICCV 2007)*, 2007, pp. 1–7.

[14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Englewood Cliffs, NJ, USA: Prentice Hall, 2002.

[15] I. Haritaoglu, D. Harwood, and L. S. Davis, "W^4: Who? when? where? what? a real time system for detecting and tracking people," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, Apr. 1998, pp. 222–227.

[16] I. Haritaoglu, D. Harwood, and L. S. Davis, "W^4: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 809–830, Aug. 2000.

[17] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, pp. 185–203, 1981.

[18] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russel, "Towards robust automatic traffic scene analysis in real-time," in *Proc. Int. Conf. Pattern Recognition*, 1994, pp. 126–131.

[19] H. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logist. Quart.*, vol. 2, no. 1–2, pp. 83–97, 1955.

[20] Z. Lin and L. S. Davis, "Shape-based human detection and segmentation via hierarchical part-template matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 604–618, 2010.

[21] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artificial Intelligence*, 1981, vol. 2, pp. 674–679.

[22] Y.-F. Ma and H.-J. Zhang, "Detecting motion object by spatio-temporal entropy," in *Proc. IEEE Int. Conf. on Multimedia and Expo*, Tokyo, Japan, Aug. 2001.

[23] S. J. McKenna and S. Gong, "Tracking colour objects using adaptive mixture models," *Image Vision Comput.*, vol. 17, pp. 225–231, 1999.

[24] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh, "Background modeling and subtraction of dynamic scenes," in *Proc. 9th IEEE Int. Conf. Computer Vision*, 2003, pp. 1305–1312.

[25] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Activity recognition using mixture of vector fields," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1712–1725, May 2013.

[26] J. C. Nascimento, J. S. Marques, and J. M. Lemos, "Modeling and classifying human activities from trajectories using a class of space-variant parametric motion fields," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 2066–2080, May 2013.

[27] J. C. Nascimento, J. S. Marques, and J. M. Lemos, "A class of space-varying parametric motion fields for human activity recognition," in *Proc. 2012 19th IEEE Int. Conf. Image Processing (ICIP)*, 2012, pp. 761–764.

[28] J. Nascimento, M. Figueiredo, and J. Marques, "Trajectory classification using switched dynamical hidden Markov models," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1338–1348, May 2010.

[29] N. Ohta, "A statistical approach to background suppression for surveillance systems," in *Proc. IEEE Int. Conf. Computer Vision*, 2001, pp. 481–486.

[30] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," *Int. J. Comput. Vision*, vol. 80, pp. 72–91, Oct. 2008.

[31] M. Seki, H. Fujiwara, and K. Sumi, "A robust background subtraction method for changing background," in *Proc. IEEE Workshop Applications of Computer Vision*, 2000, pp. 207–213.

[32] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2007 (CVPR '07)*, 2007, pp. 1–8.

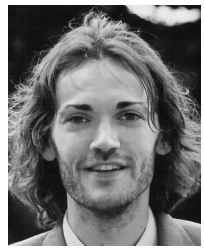
[33] N. T. Siebel and S. J. Maybank, "Real-time tracking of pedestrians and vehicles," in *Proc. IEEE Workshop Performance Evaluation of Tracking and Surveillance*, 2001.

[34] E. P. Simoncelli, "Course-to-fine estimation of visual motion," in *Proc. IEEE 8th Workshop Image and Multidimensional Signal Processing*, 1993.

[35] R. Souvenir, J. Wright, and R. Pless, "Spatio-temporal detection and isolation: Results on the PETS2005 datasets," in *Proc. IEEE Workshop Performance Evaluation in Tracking and Surveillance*, 2005.

[36] C. Stauffer, W. Eric, and L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.
[37] H. Sun, T. Feng, and T. Tan, "Spatio-temporal segmentation for video surveillance," in *Proc. IEEE Int. Conf. Pattern Recognition*, Barcelona, Spain, Sep. 2000, vol. 1, pp. 843–846.
[38] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 11, pp. 1473–1488, Nov. 2008.
[39] C. Veenman, M. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 1, pp. 54–72, Jan. 2001.
[40] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, Jul. 1997.
[41] J. Zhong and S. Sclaroff, "Segmenting foreground objects from a dynamic, textured background via a robust Kalman filter," in *Proc. 9th IEEE Int. Conf. Computer Vision*, 2003, pp. 44–50.

Manya V. Afonso (S'10–M'11) received the B.E. degree in Electronics and Telecommunication Engineering from Goa University in 2003, M.Tech. in Communication Engineering from the Indian Institute of Technology Delhi in 2005, and Ph.D. from Instituto Superior Técnico (IST), Technical University of Lisbon in 2011. He is currently a post-doctoral researcher at the Institute for Systems and Robotics (ISR) at IST. His current research interests include image processing and analysis, inverse problems, statistical inference, optimization, surveillance, and biomedical image analysis. He had previously worked in industry as a software developer in the mobile telecommunications sector.



Jacinto C. Nascimento (S'00–M'06) received the E.E. degree from Instituto Superior de Engenharia de Lisboa, in 1995, the M.Sc. and Ph.D. degrees from Instituto Superior Técnico (IST), Technical University of Lisbon, in 1998, and 2003, respectively. Currently, he is a principal researcher of a FCT project with the Institute for Systems and Robotics (ISR) at IST. Dr. Nascimento has published over 30 publications in international journals (many of which of the IEEE), has served on program committees of many international conferences, and has been a reviewer for several international journals. His research interests include statistical image processing, pattern recognition, machine learning, medical imaging analysis, video surveillance, and general visual object classification.

Jorge S. Marques received the E.E. and Ph.D. degrees, and the aggregation title from the Technical University of Lisbon, Portugal, in 1981, 1990, and 2002, respectively. Currently, he is an Associate Professor with the Electrical and Computer Engineering Department, Instituto Superior Técnico, Lisbon, and a Researcher at the Institute for Systems and Robotics. He has published over 150 papers in international journals and conferences and he is the author of the book Pattern Recognition: Statistical and Neural Methods (IST Press, 2005, 2nd ed., in Portuguese). Dr. Marques was the Co-Chairman of the IAPR Conference IbPRIA 2005, President of the Portuguese Association for Pattern Recognition (2001–2003) and Associate Editor of the Statistics and Computing Journal, Springer. His research interests are in the areas of statistical image processing, shape analysis, and pattern recognition.