Activity Recognition Using A Mixture of Vector Fields

Jacinto C. Nascimento, *Member, IEEE*, Mário A. T. Figueiredo, *Fellow, IEEE*, and Jorge S. Marques

***Abstract*— The analysis of moving objects in image sequences (video) has been one of the major themes in computer vision. In this paper, we focus on video-surveillance tasks; more specifically, we consider pedestrian trajectories and propose modeling them through a small set of motion/vector fields together with a space-varying switching mechanism. Despite the diversity of motion patterns that can occur in a given scene, we show that it is often possible to find a relatively small number of typical behaviors, and model each of these behaviors by a "simple" motion field. We increase the expressiveness of the formulation by allowing the trajectories to switch from one motion field to another, in a space-dependent manner. We present an expectation-maximization algorithm to learn all the parameters of the model, and apply it to trajectory classification tasks. Experiments with both synthetic and real data support the claims about the performance of the proposed approach.**

***Index Terms*— Expectation-maximization (EM) algorithm, human motion analysis, model selection, video surveillance.**

I. Introduction

THERE is an ongoing effort to integrate computers in several human activities, which requires computers to "understand" what people are doing and take adequate actions [1]–[3]. A standard example is video surveillance [4], [5], where the typical goal is to monitor people in indoor or outdoor environments and detect abnormal events, such fighting or moving in forbidden directions or into prohibited areas. These systems require the ability, not only to segment and track people in video sequences (already a challenging task), but also to characterize and recognize their activities.

A large fraction of the recent work on human activity recognition is based on image acquisition at short range (camera close to people) and aims at providing accurate descriptions of gestures, pose, or gait. This can be done, *e.g.*, by tracking the head, limbs, hands, torso, and by using articulated body models. Action recognition is thus performed in the space of the articulatory model parameters. Excellent surveys of this type of work can be found in [4], [6]–[8].

Manuscript received June 15, 2011; revised April 9, 2012; accepted October 10, 2012. Date of publication November 12, 2012; date of current version March 11, 2013. This work was supported by FCT project PEst-OE/EEI/LA0009/2011, by project "ARGUS" - PTDC/EEA-CRO/098550/2008. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua.

J. C. Nascimento and J. S. Marques are with the Instituto de Sistemas e Robótica, Instituto Superior Técnico, Lisbon 1049-001, Portugal (e-mail: jan@isr.ist.utl.pt; jsm@isr.ist.utl.pt).

M. A. T. Figueiredo is with the Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon 1049-001, Portugal (e-mail: mario.figueiredo@lx.it.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/TIP.2012.2226899

In the so-called far field scenario, people are far from the camera, making it impossible to obtain detailed shape information, and the systems are limited to extracting trajectories and, sometimes, a coarse shape description (*e.g.*, a bounding box, a coarse silhouette) [9]. Models of typical trajectories (motion patterns) may be estimated from training sets, and then used to classify observed trajectories/behaviors. This is a key problem in outdoor surveillance systems and is the main focus of this paper, which presents a novel approach to model human trajectories in video sequences.

In many scenarios, people tend to follow a set of typical trajectories, at least during short periods of time. Based on this observation, we propose modeling the trajectories via a small set of vector/motion fields, estimated from observed trajectory data. To increase the expressiveness of the model, we let each trajectory be split into a sequence of segments, each of which generated by one vector field. Switching between models can occur at any point in the image domain but with probability that may depend on the spatial location. This provides a flexible tool to represent a wide variety of motion patterns.

We present an expectation-maximization (EM) algorithm to learn the proposed model from sets of observed trajectories. Aiming at using the model for trajectory classification, we also consider a discriminative model selection criterion [10] to select the number of motion fields representing each class.

The remaining sections of the paper are organized as follows. Section II describes related work concerning human activity recognition and summarizes our contributions. Section III presents the proposed generative model, while Section IV describes the EM algorithm learning the model from data. The model selection procedure is presented in Section V, while Section VI reports the experimental evaluation for activity recognition in three different surveillance scenarios. Finally, Section VIII draws conclusions and discusses future work.

II. Previous Work and Contributions

A. Activity Recognition

Challenging activity recognition problems arise in several video analysis systems, where the goal is usually to track people and interpret/classify their activities. Such systems are found in applications such as intelligent environments [1], [3], human machine interaction [2], surveillance [4], and sports analysis [11]. Most of the work in this area falls into one of two different settings: short range (SR), where the camera is close to the observed people, thus detailed information of human gestures, pose, gait (and other features) can be

extracted; far field (FF), where the camera covers a wide area, thus no longer able to acquire that type of detailed information.

B. Short Range Scenario

In the SR case, several types of features have been used to characterize the human body (or part of it). Typical choices are the head and hands positions [12], appearance models [13], texture [14], blobs [15], skeletons [16], or silhouettes [17]. Some systems use articulated human body models, taking into account geometric/motion restrictions [7], [8], [18]. Activities are then represented by time evolutions in a space of articulatory features, which are often modeled using generative models, such as hidden Markov models (HMMs) [19], or variants thereof (*e.g.*, hierarchical Markov models, hidden semi-Markov models, abstract Markov models, or dynamical Bayesian networks (DBN) [20]–[24]). These models can be learned from data and can then be used to build activity classification/recognition systems.

C. Spatiotemporal Features

If dynamical information is integrated in the extracted features (yielding spatiotemporal features), there is no need for dynamic generative models, since the features already carry temporal information. These methods operate directly on a video "volume," *i.e.*, a stack of images, and have recently became popular for activity recognition [25]–[30]. A recent spatiotemporal approach uses mixtures of dynamic textures to perform clustering of the video volume [31]. More specifically, a collection of video patches is modeled as samples from set of underlying dynamic textures; application to surveillance, *e.g.*, clustering of highway traffic video, was successfully illustrated. Other examples of this class of approaches are available in the literature, *e.g*, space-time shapes [32], [33], video words [34], and 4D action features [35].

D. Far Field (FF) Scenario

In FF scenarios, it is usually impossible to obtain detailed descriptions of the observed persons, thus most methods rely solely on trajectories (*e.g.*, of the "center of mass" of the persons) obtained by tracking algorithms. Several trajectory analysis problems (such as classification and clustering) have been addressed using pairwise (dis)similarity mesures between trajectories; these include Euclidean [36] and Hausdorff distances [37], [38]. Because trajectories may have different lengths, sequence alignment techniques (*e.g.*, dynamic time warping [39] or longest common subsequence [40]) have been used to allow meaningful comparisons.

The class of approaches adopted in this paper models the trajectories as being produced by a probabilistic generative mechanism, usually an HMM or one of its variants [20]–[24]. These approaches have the key advantage of not requiring trajectory alignment or registration; moreover, they allow building a solid probabilistic inference formulation, based on which model parameters may be obtained from observed data. In that same class of approaches, Berclaz *et al* proposed a set of *behavioral maps* based on Markovian trajectory models [41]; however, their application context is orthogonal to ours, since their goal is to improve tracking results by reconstructing full trajectories from fragments thereof.

E. Recognition in the Wild

Although context was first proposed for the interpretation of static images [42]–[44], its use in human activity recognition has been rapidly expanding. Scene context has been exploited for event recognition and for the recognition of objects and actions in video [45]–[48], based on the idea that video understanding should integrate simultaneously the interpretation of objects, scenes, and actions [49]. These approaches, often known as "recognition in the wild," still face challenging problems, since the video is acquired in an uncontrolled fashion. As such, video collections often contain cluttered background, camera motion, changes in object appearance, scale, and illumination, making good features difficult to extract [50]. The dataset considered in [51] includes large variations in camera motion, object appearance, pose, and scale, as well as varying illumination conditions. More recently, the so-called *dense trajectory models* have also been used for action classification [52]; in that work, the trajectories are obtained by tracking densely sampled points at multiple spatial scales using optical flow fields.

F. Contributions

This paper introduces a novel approach to modeling trajectories in video sequences. We characterize the scene by a set of underlying motion vector fields. Each trajectory is then described as a set of consecutive segments, each of which is generated/driven by one of these underlying vector fields. Switching between fields can occur at any point in the image, following a probabilistic mechanism which can be space-dependent; *i.e.*, we have a field of switching (stochastic) matrices. This approach is flexible enough to represent a wide variety of trajectories and allows modeling space-varying behaviors without resorting to non-linear dynamical models, which are infamously hard to estimate from training data.

We also address the problem of estimating the models (*i.e.*, the velocity and switching fields) from training data. Given a set of typical trajectories, the goal is to estimate a set of vector fields and a switching field that explain well all the trajectories. Of course, this is an ill-posed inverse problem, requiring regularization; for this purpose, we assume that the motion fields are "simple" (or "smooth"). Specifically, we penalize large local changes of velocity using Gaussian field priors for the motion fields. Estimation is carried out using an EM algorithm, where the label of the active motion field of each trajectory at each time instant is treated as missing data. Experiments on both synthetic and real data show that the proposed approach is able to model a wide range of trajectories. The experiments also confirm the ability of the proposed EM algorithm to estimate the velocity fields and the switching field from observed trajectories.

Finally, we use the proposed model for trajectory classification, and test the approach on both synthetic and real data.

III. Generative Motion Model

A. Multiple Vector Fields

We assume that objects may have unconstraint motion in the image domain. We also ignore the discrete nature of digital images and model the object position at time t by a vector $\mathbf{x}_t = [x_t^1, x_t^2]^T \in \mathbb{R}^2$. As explained above, the object motion depends on its position, thus we model it using a motion field *i.e.*, we associate to each position $\mathbf{x}_t$ in the image domain a displacement vector $\mathbf{v}_t$ which depends on $\mathbf{x}_t$. Furthermore, rather than one motion field, we consider a collection of motion fields, one of which is active at each time instant.

We denote the set of vector motion fields as $\mathcal{T} = \{\mathbf{T}_1, \ldots, \mathbf{T}_K\}$, with $\mathbf{T}_k : \mathbb{R}^2 \to \mathbb{R}^2$, for $k \in \{1, \ldots, K\}$, be a set of K vector (velocity, *i.e.*, displacement in one time unit) fields. The velocity vector at point $\mathbf{x} \in \mathbb{R}^2$ of the k-th field is denoted as $\mathbf{T}_k(\mathbf{x})$. At each time instant, one of these velocity fields is *active*, *i.e.*, is driving the motion. Formally, each object trajectory is thus generated according to

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

where $k_t \in \{1, \ldots, K\}$ is the label of the active field at time t, $\mathbf{w}_t \sim \mathcal{N}(0, \sigma_{k_t}^2 \mathbf{I})$ is white Gaussian noise with zero mean and variance $\sigma_{k_t}^2$ (which may be different for each field), and L is the length (number of points) in the trajectory. The initial position follows some distribution $p(\mathbf{x}_1)$. The conditional probability density of a trajectory $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_L)$, given the sequence of active models $\mathbf{k} = \{k_1, \ldots, k_L\}$ is

$$\begin{aligned} p(\mathbf{x}|\mathbf{k}, \mathcal{T}) &= p(\mathbf{x}_1) \prod_{t=2}^{L} p(\mathbf{x}_t|\mathbf{x}_{t-1}, k_t) \\ &= p(\mathbf{x}_1) \prod_{t=2}^{L} \mathcal{N}(\mathbf{x}_t|\mathbf{x}_{t-1} + \mathbf{T}_{k_t}(\mathbf{x}_{t-1}), \sigma_{k_t}^2 \mathbf{I}) \end{aligned} \tag{2}$$

where $\mathcal{N}(\mathbf{v}|\mu, \mathbf{C})$ denotes a Gaussian density of mean μ and covariance $\mathbf{C}$, computed at $\mathbf{v}$.

The sequence of active fields $\mathbf{k} = \{k_1, \ldots, k_L\}$ is modeled as a realization of a first order Markov process, with some initial distribution $P(k_1)$, and a space-varying transition matrix which depends on the activity, *i.e.*, $P(k_t = j|k_{t-1} = i, \mathbf{x}_{t-1}) = \mathbf{B}_{ij}(\mathbf{x}_{t-1})$, where $\mathbf{B} : \mathbb{R}^2 \to \mathbb{R}^{K \times K}$ is a field of stochastic matrices

$$\mathbf{B}(\mathbf{x}) = \begin{bmatrix} B_{1,1}(\mathbf{x}) & \cdots & B_{1,K}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ B_{K,1}(\mathbf{x}) & \cdots & B_{K,K}(\mathbf{x}) \end{bmatrix}. \tag{3}$$

This model allows the switching probability to depend on the location of the object. The matrix-valued field $\mathbf{B}$ can also be seen as a set of K^2 fields with values in $[0, 1]$, under the constraint that $\sum_j B_{ij}(\mathbf{x}) = 1$, for any $\mathbf{x}$ and any i.

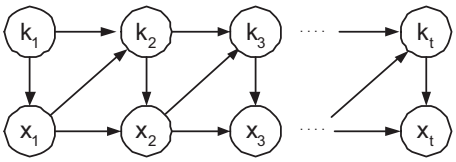

Fig. 1. Graphical model of the trajectory generation process.

The joint distribution of a trajectory and the underlying sequence of active regions, is given by

$$\begin{aligned} p(\mathbf{x}, \mathbf{k}|\mathcal{T}, \mathbf{B}) &= p(\mathbf{x}_1) P(k_1) \prod_{t=2}^{L} p(\mathbf{x}_t, k_t|\mathbf{x}_{t-1}, k_{t-1}) \\ &= p(\mathbf{x}_1) P(k_1) \prod_{t=2}^{L} \mathcal{N}(\mathbf{x}_t|\mathbf{x}_{t-1} + \mathbf{T}_{k_t}(\mathbf{x}_{t-1}), \sigma_{k_t}^2 \mathbf{I}) \\ &\quad \times B_{k_{t-1}, k_t}(\mathbf{x}_t) \end{aligned}$$

A graphical model of our generative process is shown in Fig. 1.

IV. Learning the Vector Fields

We now address the problem of learning the set of velocity fields $\mathcal{T}$, the noise variances $\boldsymbol{\sigma} = \{\sigma_1^2, \ldots, \sigma_K^2\}$, and the field of transition matrices $\mathbf{B}$ from a set of observed trajectories. Consider a training set of S independent trajectories $\mathcal{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(S)}\}$, where $\mathbf{x}^{(j)} = (\mathbf{x}_1^{(j)}, \ldots, \mathbf{x}_{L_j}^{(j)})$ is the j-th trajectory, which has length L_j. Naturally, we assume that the corresponding set of sequences of active fields, $\mathcal{K} = \{\mathbf{k}^{(1)}, \ldots, \mathbf{k}^{(S)}\}$, is not observed (it's missing). We denote the complete set of model parameters as $\theta = (\mathcal{T}, \mathbf{B}, \sigma)$.

In a classification context, we have different activity classes $\{1, \ldots, A\}$, and are given a set of training trajectories from each of these classes: $\mathcal{X}^1, \ldots, \mathcal{X}^A$. We denote the set of fields and parameters corresponding to each class a as $\theta^a = (\mathcal{T}^a, \mathbf{B}^a, \sigma^a)$, for $a = 1, \ldots, A$. In some cases, one or more of these collections of parameters may be shared among the classes; *e.g.*, if the motion fields are common among the classes and only the switching matrices differ, we have $\mathcal{T}^a = \mathcal{T}$ and $\boldsymbol{\sigma}^a = \boldsymbol{\sigma}$, for $a = 1, \ldots, A$.

A. Model Estimation Criterion: Marginal MAP (MMAP)

We now considering the problem of estimating the parameters for one of the classes, we omit the class superscript a to keep a lighter notation. The fact that the active field labels $\mathcal{K}$ are missing suggests the use of the EM algorithm [53] to find a *marginal maximum a posteriori* (MMAP) estimate of θ under some prior $p(\theta) = p(\mathcal{T})p(\mathbf{B})p(\sigma)$; formally, the estimate is given by

$$\begin{aligned} \widehat{\theta} &= \arg\max_{\theta} p(\mathbf{X}, \theta)\, p(\theta) \\ &= \arg\max_{\theta} p(\theta) \sum_{\mathcal{K}} p(\mathbf{X}, \mathcal{K}|\theta) \\ &= \arg\max_{\theta} p(\theta) \sum_{\mathcal{K}} \prod_{j=1}^{S} p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\theta) \end{aligned} \tag{4}$$

where each factor $p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\theta)$ has the form given in (4), and the sum over $\mathcal{K}$ has $K^{(\sum_j L_j)}$ terms.

Next, we will describe the E and M steps of the EM algorithm for the problem (4). For simplicity, we assume that the initial distributions $p(\mathbf{x}_1)$ and $P(k_1)$ are known; extending the algorithm to also estimate these distributions is trivial.

B. Complete Log-Likelihood

The complete log-likelihood is given by

$$\log p(\mathcal{X}, \mathcal{K}|\theta) = \sum_{j=1}^{S} \log p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\theta) \tag{5}$$

where each term $p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\theta)$ has the form (see (4))

$$\begin{aligned} p(\mathbf{x}^{(j)}, \mathbf{k}^{(j)}|\theta) = {} & p(\mathbf{x}_1^{(j)})\, P(\mathbf{k}_1^{(j)}) \\ & \times \prod_{t=2}^{L_j} \mathcal{N}(\mathbf{x}_t^{(j)}|\mathbf{x}_{t-1}^{(j)} + \mathbf{T}_{k_t^{(j)}}(\mathbf{x}_{t-1}^{(j)}), \sigma^2_{k_{t-1}^{(j)}}\mathbf{I}) \\ & \times B_{k_{t-1}^{(j)}, k_t^{(j)}}(\mathbf{x}_{t-1}^{(j)}). \end{aligned} \tag{6}$$

Let the missing model labels be represented by binary indicator variables: each label $k_t^{(j)} \in \{1, \ldots, K\}$ (the active field at time t of trajectory j) is represented by a binary vector $\mathbf{y}_t^{(j)} = (y_{t,1}^{(j)}, \ldots, y_{t,K}^{(j)}) \in \{0,1\}^K$, where $y_{t,l}^{(j)} = 1 \Leftrightarrow k_t^{(j)} = l$. Using this notation, the complete log-likelihood becomes

$$\begin{aligned} & \log p(\mathcal{X}, \mathcal{K}|\theta) \\ & = \sum_{j=1}^{S}\sum_{t=2}^{L_j}\sum_{l=1}^{K}\sum_{g=1}^{K} y_{t-1,g}^{(j)}\, y_{t,l}^{(j)} \log B_{g,l}(\mathbf{x}_{t-1}^{(j)}) \\ & + \sum_{j=1}^{S}\sum_{t=2}^{L_j}\sum_{l=1}^{K} y_{t,l}^{(j)} \log \mathcal{N}(\mathbf{x}_t^{(j)}|\mathbf{x}_{t-1}^{(j)} + \mathbf{T}_l(\mathbf{x}_{t-1}^{(j)}), \sigma_l^2\mathbf{I}) + C, \end{aligned} \tag{7}$$

where $C = \sum_{j=1}^{S} \log p(\mathbf{x}_1^{(j)}) + \log P(k_1^{(s)})$ is an irrelevant constant, which will be dropped in the sequel.

C. E-Step

Note that $\log p(\mathcal{X}, \mathcal{K}|\theta)$ is linear with respect to the missing binary indicators $y_{t,l}^{(j)}$ and to products of pairs of consecutive binary indicators $y_{t-1,g}^{(j)}\, y_{t,l}^{(j)} \equiv s_{t,g,l}^{(j)}$; these products are switching indicators, since $s_{t,g,l}^{(j)} = 1$ if and only if trajectory j switched from field g, at time $t-1$, to field l at time t. The consequence of this linearity is that computing the conditional expectation of the complete log-likelihood reduces to computing the conditional expectations of these indicator variables, denoted as $\bar{y}_{t,l}^{(j)}$ and $\bar{s}_{t,g,l}^{(j)}$, given the observed trajectories and current estimate $\widehat{\theta}$, and then plugging them into the complete log-likelihood. The so called Q-function is thus given by

$$\begin{aligned} Q(\theta\widehat{\theta}) & \equiv \mathbb{E}\left[\log p(\mathcal{X}, \mathcal{K}|\theta)|\mathcal{X}, \widehat{\theta}\right] \\ & = \sum_{j=1}^{S}\sum_{t=2}^{L_j}\sum_{l=1}^{K} \bar{y}_{t,l}^{(j)} \log \mathcal{N}(\mathbf{x}_t^{(j)}|\mathbf{x}_{t-1}^{(j)} + \mathbf{T}_l(\mathbf{x}_{t-1}^{(j)}), \sigma_l^2\mathbf{I}) \\ & \quad + \sum_{j=1}^{S}\sum_{t=2}^{L_j}\sum_{l=1}^{K}\sum_{g=1}^{K} \bar{s}_{t,g,l}^{(j)} \log B_{g,l}(\mathbf{x}_{t-1}^{(j)}), \end{aligned} \tag{8}$$

where (because $y_{t,l}^{(j)}$ and $s_{t,g,l}^{(j)}$ are binary)

$$\bar{y}_{t,l}^{(j)} = \mathbb{P}\left[y_{t,l}^{(j)} = 1 \middle| \mathbf{x}^{(j)}, \widehat{\theta}\right] \tag{9}$$

$$\bar{s}_{t,g,l}^{(j)} = \mathbb{P}\left[s_{t,g,l}^{(j)} = 1 \middle| \mathbf{x}^{(j)}, \widehat{\theta}\right]. \tag{10}$$

These probabilities are obtained by a forward-backward procedure [19], modified to take into account that the transition matrix is not constant, but depends on the trajectories.

D. M-Step

In the M-step, the estimates are updated according to

$$\widehat{\theta}_{\text{new}} = \arg\max_{\theta} Q(\theta; \widehat{\theta}) + \log p(\theta). \tag{11}$$

In this section addresses this maximization in detail, as well as the adopted priors, by looking separately at the maximization with respect to each component of $\theta = (\mathcal{T}, \mathbf{B}, \sigma)$.

1) Updating $\widehat{\sigma}$: We adopt flat priors for σ, *i.e.*, we use maximum likelihood noise variance estimates. Computing the partial derivative of $Q(\theta; \widehat{\theta})$ with respect to each component σ_k^2 of σ, and equating to zero, we obtain (for $k = 1, \ldots, K$)

$$(\widehat{\sigma}_k^2)_{\text{new}} = \frac{\displaystyle\sum_{j=1}^{S}\sum_{t=2}^{L_j} \bar{y}_{t,k}^{(j)}\, \|\mathbf{x}_t^{(j)} - \mathbf{x}_{t-1}^{(j)} - \mathbf{T}_k(\mathbf{x}_{t-1}^{(j)})\|^2}{\displaystyle\sum_{j=1}^{S}\sum_{t=2}^{L_j} \bar{y}_{t,k}^{(j)}}.$$

2) Updating $\widehat{\mathcal{T}}$: As mentioned in the introduction, estimating the motion fields requires regularization. Moreover, to avoid optimization with respect to infinite dimensional objects, we adopt a finite dimensional parametrization, where each field is a linear combination of basis functions/fields, *i.e.*

$$\mathbf{T}_k(\mathbf{x}) = \sum_{n=1}^{N} \mathbf{t}_k^{(n)}\, \phi_n(\mathbf{x}), \tag{12}$$

where $\phi_n(\mathbf{x}) : \mathbb{R}^2 \to \mathbb{R}$, for $n = 1, \ldots, N$, is a set of basis functions (scalar basis fields), with ϕ_n centered at the node $\mathbf{u}_n = [u_n^1, u_n^2]^T$. Although more sophisticated basis functions could be used, we adopt simple bilinear interpolation, thus

$$\phi_n(\mathbf{x}) = \begin{cases} \delta^{-2}\left(|x^1 - u_n^1| \cdot |x^2 - u_n^2|\right) & \text{if } \|\mathbf{x} - \mathbf{u}_n\|_\infty < \delta \\ 0 & \text{otherwise,} \end{cases}$$

and each $\mathbf{t}_k^{(n)} \in \mathbb{R}^2$. Collecting all these vector coefficients in $\tau_k \in \mathbb{R}^{2N\times 1}$, defined according to

$$\tau_k = \left[(\mathbf{t}_k^{(1)})^T, \ldots, (\mathbf{t}_k^{(N)})^T\right]^T \tag{13}$$

and letting

$$\Phi(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) & 0 & \phi_2(\mathbf{x}) & 0 & \cdots & \phi_N(\mathbf{x}) & 0 \\ 0 & \phi_1(\mathbf{x}) & 0 & \phi_2(\mathbf{x}) & \cdots & 0 & \phi_N(\mathbf{x}) \end{bmatrix}, \tag{14}$$

we can write

$$\mathbf{T}_k(\mathbf{x}) = \Phi(\mathbf{x})\, \tau_k. \tag{15}$$

Thus, estimating the field $\mathbf{T}_k$ becomes equivalent to estimating the coefficient vector τ_k.

In this paper, we assume that the motion fields are smooth, *i.e.*, that abrupt velocity changes should be modeled by a switching event, not by non-smoothness in the motion fields. Consider a neighborhood system in the image grid (*e.g.*, 4-connected neighborhood) and let i and j be two neighboring nodes. Denote as $\mathbf{d}_k^{(i,j)} = \mathbf{t}_k^{(i)} - \mathbf{t}_k^{(j)}$ the motion change between two neighboring nodes and Δ_k the vector of all the velocity differences for the k-th motion field the vector Δ_k can be wtrtten as

$$\Delta_k = \Gamma \tau_k \tag{16}$$

where Γ is a sparse matrix which computes the velocity differences between all the pairs of neighboring. We assume that Δ_k follows a Gaussian prior, $\mathcal{N}(0, \alpha^2 \mathbf{I})$, which therefore induces an also Gaussian prior on the coefficients

$$p(\tau_k) \propto \exp\{-\frac{1}{2\,\alpha^2}\ \tau_k^T\ \Gamma^T \Gamma \tau_k\}, \tag{17}$$

where α^2 can be seen as a global variance factor that controls the "strength" of the prior (low value corresponds to a strong prior), while the shape of the inverse covariance $\Gamma^T \Gamma$ depends on the basis functions ϕ_i; for more details on this type of Gaussian field priors, see [54].

The term of $Q(\theta; \widehat{\theta}) + \log p(\tau_k)$ that depends on τ_k (apart from a constant) is equal to

$$-\frac{1}{2\,\alpha^2}\tau_k^T\ \Gamma^T \Gamma \tau_k - \sum_{j=1}^{S}\sum_{t=2}^{L_j} \bar{y}_{t,k}^{(j)} \Bigg[\log(2\pi\,\sigma_k^2) + \frac{1}{2\sigma_k^2}\,\|\mathbf{x}_t^{(j)} - \mathbf{x}_{t-1}^{(j)} - \Phi(\mathbf{x}_{t-1}^{(j)})\tau_k\|^2 \Bigg]. \tag{18}$$

Computing the gradient w.r.t. τ_k and equating to zero, leads to a pair of linear system of equations

$$\left(\mathbf{R}_k + \frac{\Gamma^T \Gamma}{\alpha^2}\right)\ \tau_k = \mathbf{r}_l \tag{19}$$

where

$$\mathbf{R}_k = \sum_{j=1}^{S}\sum_{t=2}^{L_j} \frac{\bar{y}_{t,k}^{(j)}}{\sigma_k^2} \left(\Phi(\mathbf{x}_{t-1}^{(j)})\right)^T \Phi(\mathbf{x}_{t-1}^{(j)}) \tag{20}$$

and

$$\mathbf{r}_k = \sum_{j=1}^{S}\sum_{t=2}^{L_j} \frac{\bar{y}_{t,k}^{(j)}}{\sigma_k^2} \left(\Phi(\mathbf{x}_{t-1}^{(j)})\right)^T (\mathbf{x}_t^{(j)} - \mathbf{x}_{t-1}^{(j)}). \tag{21}$$

Notice that since $\Phi(\mathbf{x}_{t-1}^{(j)})$ is $2 \times 2N$ (see (14)), matrix $\mathbf{R}_l$ is $2N \times 2N$ and $\mathbf{r}_l$ is $2N \times 1$ (as is τ_k). Solving (19), yields $(\widehat{\tau_k})_{\text{new}}$, for $k = 1, \ldots, K$, which in turn define $\widehat{\mathcal{T}}_{\text{new}} = (\widehat{\mathbf{T}}_1, \ldots, \widehat{\mathbf{T}}_K)_{\text{new}}$.

3) Updating $\widehat{\mathbf{B}}$*:* We follow the same strategy adopted for the motion fields, *i.e.*, we represent this field on a set of scalar basis functions, $\psi_i(\mathbf{x}) : \mathbb{R}^2 \to \mathbb{R}$, for $m = 1, \ldots, M$

$$\mathbf{B}(\mathbf{x}) = \sum_{m=1}^{M} \mathbf{b}^{(m)}\ \psi_m(\mathbf{x}) \tag{22}$$

where each $\mathbf{b}^{(m)} \in \mathbb{R}^{K \times K}$ is itself a stochastic matrix, *i.e.*, for any $m = 1, \ldots, M$, and any $p = 1, \ldots, K$

$$\sum_{k=1}^{K} b_{p,k}^{(m)} = 1. \tag{23}$$

Of course, it must be guaranteed that this representation yields a stochastic matrix at any location $\mathbf{x}$. A sufficient condition for all the entries of the expansion to be non-negative is that $\psi_m(\mathbf{x}) \geq 0$, for all $\mathbf{x}$ and all $m = 1, \ldots, M$. Moreover, since each $\mathbf{b}^{(m)}$ is a stochastic matrix

$$1 = \sum_{k=1}^{K}\sum_{m=1}^{M} b_{p,k}^{(m)}\ \psi_m(\mathbf{x}) = \sum_{m=1}^{M} \psi_m(\mathbf{x}) \sum_{k=1}^{K} b_{p,k}^{(m)} = \sum_{m=1}^{M} \psi_m(\mathbf{x}).$$

Consequently, a sufficient condition for $\mathbf{B}(\mathbf{x})$ to be a stochastic matrix is that the basis functions verify the so-called partition of unity: at any point $\mathbf{x}$, $\psi_i(\mathbf{x}) \geq 0$, for all m, and

$$\sum_{m=1}^{M} \psi_m(\mathbf{x}) = 1;$$

these condition are satisfied by bilinear interpolating functions (and, more generally, by any B-spline basis functions [55]).

Given the representation (22), the problem of estimating $\mathbf{B}$ becomes that of estimating an M-tuple of stochastic matrices $\beta = (\mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(M)})$, by maximizing $Q(\theta; \widehat{\theta})$, under the constraint (23). Inserting (22) into (8), and dropping all constant terms, the objective function can be written as

$$\mathcal{E}(\beta) = \sum_{j=1}^{S}\sum_{t=2}^{L_j}\sum_{l=1}^{K}\sum_{g=1}^{K} \bar{s}_{t,g,l}^{(j)} \log \sum_{m=1}^{M} b_{g,l}^{(m)}\ \psi_m(\mathbf{x}_{t-1}^{(j)}), \tag{24}$$

which should be maximized under the constraint in (23). We attack this constrained problem using the gradient projection (GP) algorithm [56].

The two ingredients of the GP algorithm are the the gradient of the objective and the projection onto the constraint set. Concerning the gradient, it is simple to compute the partial derivatives of $\mathcal{E}(\beta)$ with respect the $b_{g,l}^{(m)}$, which are given by

$$\frac{\partial \mathcal{E}(\beta)}{\partial b_{g,l}^{(m)}} = \sum_{j=1}^{S}\sum_{t=2}^{L_j} \bar{s}_{t,g,l}^{(j)}\ \frac{\psi_m(\mathbf{x}_{t-1}^{(j)})}{B_{g,l}(\mathbf{x}_{t-1}^{(j)})}.$$

Concerning the projection of a matrix onto the set of stochastic matrices, it is equivalent to projecting each row onto to the probability simplex, for which we use a recently proposed fast (*i.e.*, with complexity $O(K \log K)$) algorithm [57].

V. Model Selection

Up to now, we have assumed that the number of fields associated to each type of activity is a priori known. In practice, of course, this needs be learned from data; we consider two scenarios where this task needs to be addressed.

In the first scenario, which we refer to as restricted, we assume that all the activities have the same complexity and share the same vector fields. This means that the class-specific models are $\theta_K^{(a)} = (\mathcal{T}, \mathbf{B}^{(a)}, \sigma)$, for $a \in \{1, \ldots, A\}$, where only the switching matrices differ among the activity classes, and K is the number of (shared) motion fields. Learning these models from training data requires a minor modification of the EM algorithm presented in the previous section: the log-likelihood term that depends on the motion field parameters are built by pooling together the trajectories of all the classes.

TABLE I
VALIDATION AND TEST ACCURACY (IN %) FOR THE RESTRICTED (TOP) AND UNRESTRICTED (BOTTOM) FORMULATIONS IN THE STRAIGHT AND SPREAD EXAMPLE

N^o of models	1	2	3	4	5
Selection set	51.20	**99.40**	95.40	99.00	**99.40**
Testing set	49.40	**99.40**	95.20	97.80	97.40

Models configuration (K_1, K_2)	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)	(3, 1)	(3, 2)	(3, 3)
Selection set	**100**	**100**	**100**	**100**	**100**	**100**	97.33	98.00	98.33
Testing set	**100**	**100**	**100**	**100**	**100**	**100**	96.75	99.50	98.75

In the second scenario, referred to as unrestricted, we assume that each activity may have a different complexity, thus possibly a different number of fields. The model for each activity/class is independently estimated from the subset of the training data from that class, *i.e.*, we independently learn class-specific models $\theta_{K_a}^{(a)} = (\mathcal{T}^{(a)}, \mathbf{B}^{(a)}, \sigma^{(a)})$, for $a \in \{1, \ldots, A\}$, where K_a is the order of the model for class a.

To determine the model complexity (*i.e.*, K in the restricted case and $(K_1, \ldots, K_A)$ in the unrestricted case), we adopt a discriminative model selection criterion [58], which aims at choosing the model that achieves the best recognition accuracy. Consider a training set of trajectories, $\mathcal{X} = \{\mathcal{X}^{(1)}, \ldots, \mathcal{X}^{(A)}\}$, where $\mathcal{X}^{(a)} = \{\mathbf{x}^{(1,a)}, \ldots, \mathbf{x}^{(S_a,a)}\}$ is the subset of training trajectories that belongs to activity class a. Assume also that, in addition to the training set $\mathcal{X}$, we have a so-called *selection* set $\mathcal{D} = (\mathcal{D}^{(1)}, \ldots, \mathcal{D}^{(A)})$, where $\mathcal{D}^{(a)} = \{\bar{\mathbf{x}}^{(1,a)}, \ldots, \bar{\mathbf{x}}^{(D_a,a)}\}$ is the subset of selection trajectories belonging to class a. Model selection is thus performed by using this set $\mathcal{D}$ to assess the classification accuracy achieved with each model order.

In the *restricted* case, we assume that $K \in \{1, \ldots, M\}$. In the *unrestricted* case, we define $K = (K_1, \ldots, K_A) \in \{1, \ldots, M\}^A$, referred to as *model configuration*, specifying the number of vector fields in the model of each activity classe.

The accuracy of a given model order/configuration, as measured on the *selection* set, is

$$\mathcal{M}(K, \mathcal{D}) = \left(\sum_{a=1}^{A} \sum_{j=1}^{D_a} \right)^{-1} \times \sum_{a=1}^{A} \sum_{j=1}^{D_a} \mathbb{I}\left[a = \arg \max_{r \in \{1,\ldots,A\}} p(\bar{\mathbf{x}}^{(j,a)} | \theta_{K_r}^{(r)}) \right], \tag{25}$$

where $\mathbb{I}$ is the indicator function, which equals 1, if the proposition in its argument is true, and zero, if it is false. The discriminative model selection criterion consists in finding the model or configuration that maximizes $\mathcal{M}(K, \mathcal{D})$. In the case of a draw, the model/configuration with fewer fields is selected.

VI. EXPERIMENTAL RESULTS

We report experimental results, using both synthetic and real data, including extensive comparisons with other methods.

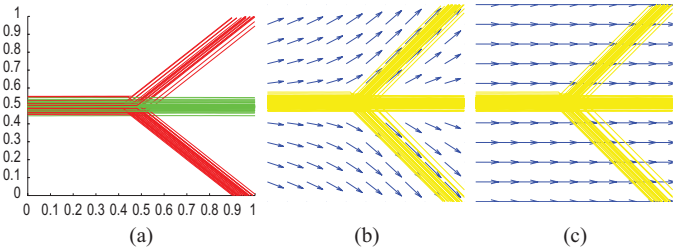

Fig. 2. First synthetic example. Two trajectory classes. (a) Green: straight. Red: spread. (b) and (c) Examples of vector field estimates obtained with the EM algorithm using ten iterations.

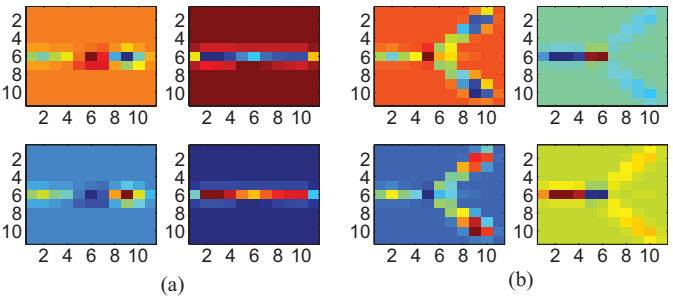

Fig. 3. Transitions matrix field estimates for activities. (a) Straight. (b) Spread. In each case, we display the fields of the four elements of the transition matrices, in their natural positions.

A. Synthetic Data

1) First Synthetic Experiment: "Straight and Spread": In this example, the trajectories are generated as follows. All trajectories start at a small region around point $[0,\ 0.5]^T$ of the unit square (Fig. 2 (a)). There are two activity classes: class 1 (called "straight," shown in green), where the trajectories are straight from left to right, with zero switching probability (identity transition matrices everywhere); class 2 (called "spread", shown in red), where in the region around the center of the square, the trajectory can turn up or down by 45°; in this region, the transition matrix has diagonal elements equal to 0.8 and the remaining ones equal to $0.2/(K-1)$.

We generate 100 training, 100 selection, and 100 testing trajectories. In the restricted formulation, we let $M = 5$, while in the unrestricted case, we set $M = 3$. To account for the tendency of EM to converge towards local minima, we consider four different random initializations and report average results. Fig. 2 (b, c) displays examples of motion field estimates

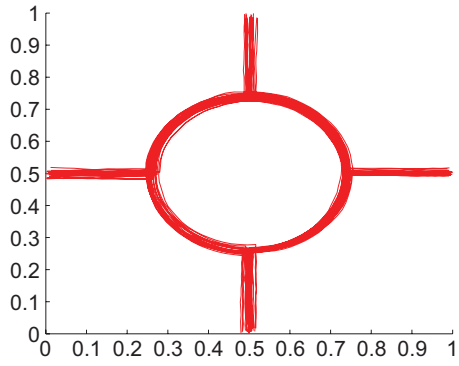

Fig. 4. Sample trajectories in the roundabout synthetic experiment. All trajectories enter through the bottom road and exit at one of the four roads.

(blue arrows) obtained by the proposed EM algorithm, together with a set of observed trajectories (yellow). Table I (top) reports classification accuracies (mean values over 4 EM runs). Naturally, with only one model per class ($K = 1$) in the restricted case, since the motion field in Fig. 2 (b) also explains quite well the straight trajectories, the classification accuracy is essentially the same as that of random guessing. With $K = 2$ shared motion fields (precisely those shown in Fig. 2 (b, c), the presence or absence of switching (expressed by the two different switching fields) is enough to achieve almost perfect accuracy. Finally, Fig. 3 shows the transition matrix field estimates.

In the unrestricted case, we have $K = (K_1, K_2) \in \{1, 2, 3\}^2$ (9 candidate configurations). Table I (bottom) shows the accuracy of each of the model configurations. In this simple toy example, several model configurations achieve perfect accuracy; since one motion field per class is enough to achieve 100% accuracy, $K = (1, 1)$ should be the chosen configuration. In this case, the motion field estimates are indistinguishable from those in Fig. 2 (b, c).

2) Second Synthetic Experiment: "Roundabout": The second synthetic example (Fig. 4) models a roundabout (where, say cars, circulate counterclockwise) connected to four roads. We consider four activities classes, shown in Fig. 5, all corresponding to cars that enter at the bottom and do one of four things: circle one quarter of the roundabout and exit at the first road; circle one half of the roundabout and exit through the second road; circle three quarters of the roundabout and exit at the third road; fully circle the roundabout and exit through the bottom road. As above, we generate 100 training, 100 validation, and 100 test trajectories, and report average results from 4 EM runs.

The values in Table II (top) show perfect accuracy is obtained for $K = 2$ and that good generalization to the test set occurs. Fig. 6 shows the vector fields estimates for $K = 2$, while Fig. 7 shows the estimates of the transition matrix fields. Clearly, these motion fields capture the outwards and circular motion patterns, showing that the proposed framework allows a compact description of quite a large range of trajectories. Discrimination among the four activity classes is carried solely by the different transition matrix fields.

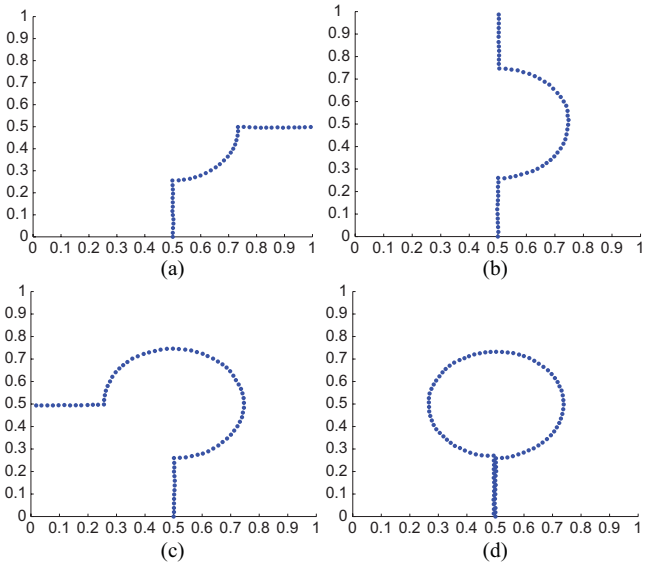

Fig. 5. Four activities considered in this roundabout scenario (see text for details).

TABLE II

VALIDATION AND TEST SET ACCURACY (IN %) FOR THE RESTRICTED (TOP) AND UNRESTRICTED (BOTTOM) FORMULATION IN THE ROUNDABOUT EXAMPLE. IN THE BOTTOM PART, THE CONFIGURATION NUMBERS CORRESPOND TO THE LEXICOGRAPHIC ORDER ($1 = (1, 1, 1, 1)$, $2 = (1, 1, 1, 2)$, $\ldots$, $16 = (2, 2, 2, 2)$)

$N°$ of models	1	2	3	4	5
Validation	24.80	**100.00**	97.40	99.20	98.20
Test	27.80	**100.00**	95.20	98.60	97.60

Models config.	1, 5, 9, 13	2, 6	3, 7	4, 8	10, 14	11, 15	12, 16
Validation	**100.00**	96.00	99.33	95.33	98.33	**100.00**	98.33
Test	**99.67**	95.67	98.00	94.33	94.33	98.67	93.67

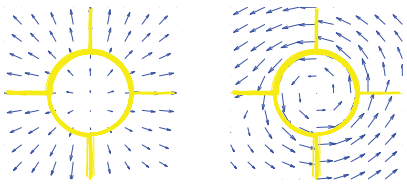

Fig. 6. Blue arrows: Vector field estimates for the restricted formulation with $K = 2$. Superimposed in yellow: several trajectories for reference.

For the unrestricted formulation (Table II, bottom), we let $K = (K_1, K_2, K_3, K_4) \in \{1, 2\}^4$ (16 possible configurations). The results in Table II (bottom) show that perfect accuracy is achieved with a single model per class. Fig. 8 depicts the estimates of the vector fields.

3) Third Synthetic Experiment: "Straight and Spread Revisited": The third set of synthetic experiments addresses two aspects: the robustness of the proposed method with respect to speed (recall that the speed is the norm of the velocity vector) variations among the trajectories and within each trajectory. These aspects are important for the effectiveness of

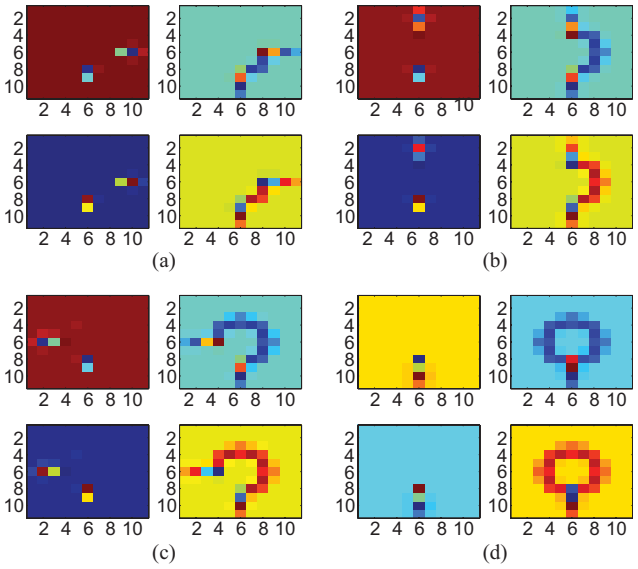

Fig. 7. Transition matrix fields estimated for the four activities in the roundabout experiment, in the restricted formulation. (a)–(d) Correspond to the activities with the same labels in Fig. 5.

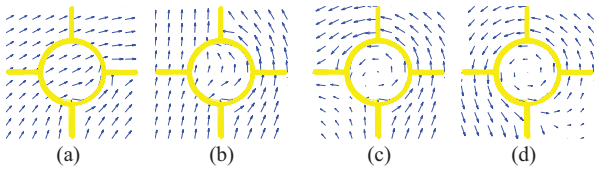

Fig. 8. Vector fields estimates (one per class, in the unrestricted formulation) for the roundabout experiment. (a)–(d) Correspond to the activities with the same labels in Fig. 5.

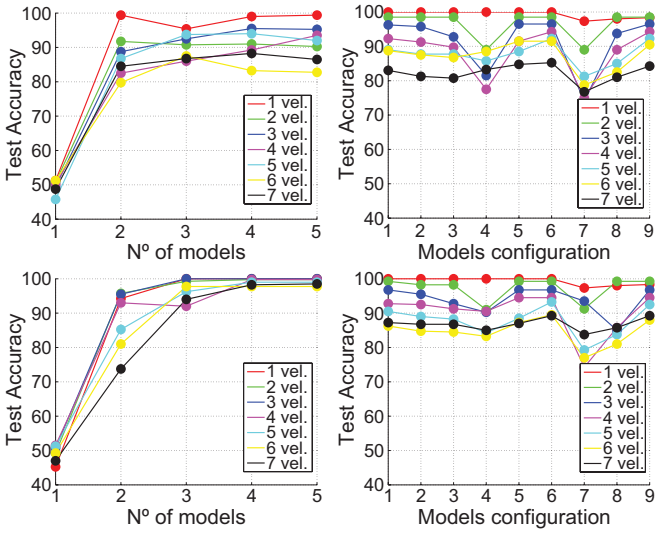

Fig. 9. Performance of the constrained versions of the algorithm (shared motion fields) when varying the velocities among and within the trajectories. Left: first experiment. Right: second experiment.

Fig. 10. Trajectories extracted in three surveillance scenarios. (a) UCSD. (b) Campus. (c) Shopping center.

the method in some real scenarios, such as when classifying activities with similar motion patterns but varying speeds.

To assess robustness with respect to speed variations among the trajectories in each class, we generate 100 trajectories per set (training, validation and test sets) from both the straight and spread activities, as in Section VI-A.1, but with a difference: the speed varies among trajectories in the set. We performed 7 experiments, each with trajectories with N_v different speeds (*i.e.* $N_v \in \{1, \ldots, 7\}$). The speed values are defined in the set $I = \{v_{\text{ini}} + (N_v - 1)\Delta\}$ for $N_v \in [1, \ldots, 7]$. Trajectories with varying speed are generated as follows: (i) obtain a random value $r \in [0, 1]$; (ii) find the subinterval such that $\frac{i-1}{N_v} < r \leq \frac{i}{N_v}$, (iii) set $\kappa = v_{\text{ini}} + (i-1)\Delta$ as a multiplicative factor applied to the velocity vector $\mathbf{T}_{k_t}(\mathbf{x}_{t-1})$ in (1). In the experiment, we set $v_{\text{ini}} = 0.5$ and $\Delta = 1.5$. This allows us to have significant speed changes among trajectories.

To assess robustness with respect to speed changes within each trajectory, we defined $N_v \in \{1, \ldots, 7\}$ possible speed values. The trajectories are generated as follows: (i) decide randomly if the speed will increase or decrease; (ii) divide the trajectory in N_v equal-length segments and increase/decrease the speed by Δ from a segment to the next one. We use a small value $\Delta = 1/3$, since it is not expected to have abrupt speed changes in far-field surveillance settings. Fig. 9 (top row) shows the results (mean accuracy over 4 EM runs) on the test set under these conditions varying the number of models from 1 up to 5 (constrained case-left panel of the figure) and from 1 up to 3 models (unconstrained case-right panel). It can be observed that by limiting the number of models, *e.g.* $K = 2$, the performance decreases as the number of different speeds present increases. However, this accuracy degradation is avoided by allowing the number of models to increase. The results for the unconstrained case (not shown for lack of space) lead to the same general conclusion.

4) Limitations of the Proposal: Of course, the proposed method is not able to classify certain types of activities. Consider, for example, the roundabout scenario, and the two following activity classes: in class 1, the cars go around once and leave through the bottom road; in class 2, the only difference is that the cars go around twice before leaving. Of course, the models estimated from trajectories of both classes would be similar, thus unable to distinguish these classes.

B. Real Data

We consider three different real surveillance scenarios, shown in Fig. 10. In this case, and since we have a limited number of trajectories, we use 5-fold cross validation (CV) on the training set to perform model selection.

1) UCSD Dataset: The UCSD dataset [59] contains 189 trajectories (Fig. 10 (a)), which we divide into two classes: "moving left" and "moving right." Fig. 11 shows the scene and

TABLE III

VALIDATION AND TEST SET ACCURACY (IN %) FOR THE RESTRICTED (TOP) AND UNRESTRICTED (BOTTOM) FORMULATIONS IN THE UCSD DATASET

N° of models	1	2	3	4	5
Validation	43.05	**100.00**	99.34	85.43	86.09
Test	42.86	**98.94**	97.88	87.30	88.89

Models configuration (K_1, K_2)	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)	(3, 1)	(3, 2)	(3, 3)
Validation	**98.01**	97.35	95.36	98.01	98.68	98.01	98.01	98.01	97.35
Test	**98.94**	98.94	97.88	97.36	97.36	97.88	96.83	96.83	98.94

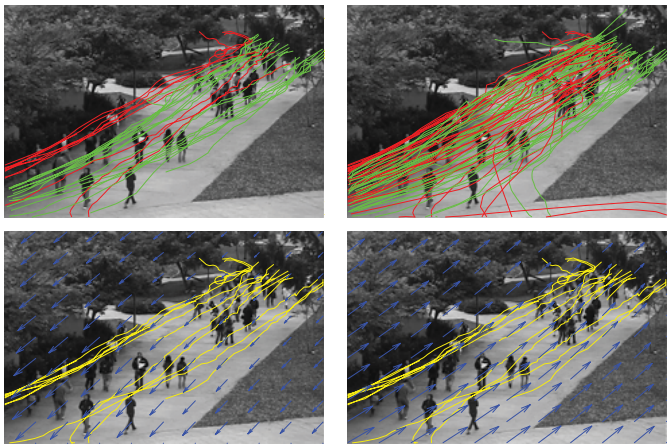

Fig. 11. UCSD experiment: two snapshots and trajectories of two different activities. Red: moving left. Green: moving right. Bottom row-Blue arrows: estimates of the vector fields for each activity. Yellow lines: superimposed with training trajectories.

a subset of the observed trajectories from both activity types (represented in red and green). The same Fig. 11 (bottom row) shows the two motion fields obtained by the EM algorithm. Notice how, although the physical velocity is approximately constant, the velocity in the image plane is lower at the top of the image than it is at the bottom, due to perspective.

Table III (top) shows the accuracy values as functions of the number of models K, in the restricted formulation, showing that, naturally, $K = 2$ is the choice yielding the best performance. Table III (bottom) shows the accuracy with the unrestricted formulation. As expected, each activity in the UCSD data set is well described by a single motion field, thus configuration $(K_1, K_2) = (1, 1)$ achieves the best accuracy.

2) University Campus: In this scenario, the images are first transformed via an homography, to compensate for the strong perspective distortion (see Fig. 12), yielding a so-called bird's eye view, which eliminates speed changes due to perspective. To obtain the trajectories, we first detect all active regions in each image and then track them using the Lehigh omnidirectional tracking system (LOTS) [60]. A pair of regions (A, B) detected in consecutive frames is associated if B is the only region in the second frame that overlaps with A and vice-versa. This can be interpreted as mutual favorite pairing [62]. The associated trajectories were manually edited to correct wrong or missing connections.

Fig. 12. Six most activity classes in the campus experiment. Red and green: the pedestrians leave and enter the building, respectively. Pink (up) and yellow (down): pedestrians walk diagonally avoiding the cars in the parking lot. Cyan: pedestrians walk directly into the building using the zebra crossing. Blue: pedestrians walk parallel to the facade of the building.

TABLE IV

VALIDATION AND TEST SET ACCURACY (IN %) FOR THE RESTRICTED FORMULATION IN THE CAMPUS DATASET

N° of models	1	2	3	4	5	6	7
Validation	14.29	48.57	91.43	92.86	95.71	**94.29**	98.57
Test	14.12	48.24	95.29	95.29	94.12	**98.82**	92.94

TABLE V

SELECTION AND TESTING ACCURACY (IN %) FOR THE RESTRICTED FORMULATION IN THE SHOPPING SCENARIO

N° of models	1	2	3	4	5	6	7
Validation	13.51	59.46	56.76	64.87	64.87	67.57	**75.68**
Test	13.04	58.70	60.87	56.52	63.04	63.04	**76.09**

Table IV shows that, although the accuracy is good for a range of values of K (except for $K = 1$), choosing higher model orders improves the accuracy, with the best result (naturally, since there are 6 classes) obtained for $K = 6$. In the unrestricted formulation, a similar behavior is observed; here, we set $M = 2$, yielding 64 different configurations. In this case, the best performance (97%) is achieved by a configuration with one field per class, $K = (1, 1, 1, 1, 1, 1)$.

3) Shopping Center: This final experiment considers video surveillance data from a shopping center, extracted from

TABLE VI
OVERALL ACCURACY OF THE PROPOSED METHOD AND THE METHOD FROM [63]; (A) AND (B) INDICATE COMMON COVARIANCE AND COMMON DIAGONA COVARIANCE, RESPECTIVELY

	Proposed		[63]	
	Constrained	Unconstrained	(A)	(B)
UCSD	99.41%	**99.62%**	98.95%	99.48%
Campus	98.77%	**100%**	99.72%	**100%**
Shopping	76.68%	85.45%	**87.39%**	82.65%

(a) (b) (c) (d) (e)

Fig. 13. Examples of trajectories belonging to client class. (a) Person enters from the left. (b) Person leaves the mall in the opposite direction. Examples of trajectories belonging to nonclient class. (c) Passing. (d) In-out$_1$. (e) In-out$_2$.

the CAVIAR[1] database (see Fig. 10(c)). As in the previous experiment, we use a homographic transformation. Two groups of trajectories are considered: client and non-client. In the first class (client), the pedestrian enters the store, stays there for a considerable amount of time, possibly out of the camera field of view. In this case, the trajectory is discontinuous. The same happens if one observes a person leaving the store, meaning that the person stayed inside for some time. These trajectories are termed herein as "entering" (Fig. 13(a)) and "leaving" (Fig. 13(b)). In the second group (non-client), the entire trajectory is observed (unless some occlusion happens, but for a short period). Fig. 13 (c, d, e) shows several trajectories in the non-client group. Fig. 13(c) shows a "passing" activity: the person passes in the front of the store. Fig. 13(d) shows the pedestrian entering the field of view (from the right), wandering into the store, and exiting towards the left; these trajectories are termed "in-out" (type 1). In Fig. 13 (e), the person enters the store (from the left) and then exits towards the left; these trajectories are termed "in-out" (type 2).

A preliminary experiment was done to assess how the accuracy depends on parameter α^2 (see (17)). The plots in Fig. 14 show that the best accuracies are achieved for an "intermediate" value. In fact, if α^2 is too high (weak smoothing), the algorithm may over-fit the noise and irrelevant details of the trajectories. On the other hand, over-smoothing (α^2 too low) is may prevent the fields from capturing the features that distinguish the different trajectory classes. Several experiments suggested that, for this scenario, $\alpha^2 = 0.1$ seems to be a good compromise, and it will be used for further tests in this scenario. Although not shown, similar results were obtained for the restricted formulation (shared fields).

To select the model configuration, we used the 5-fold cross validation procedure mentioned above, using four different EM initializations (with $\alpha^2 = 0.1$ fixed). Fig. 15 shows that the best accuracies are by model configuration with more than one model per activity.

[1]http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/

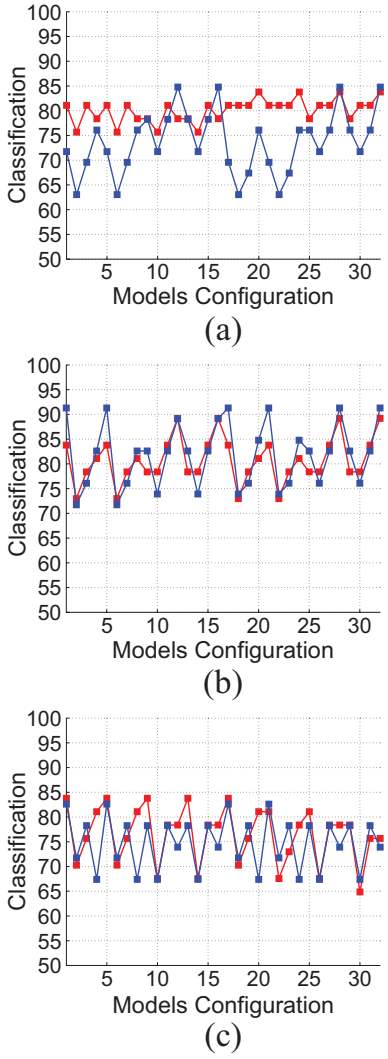

Fig. 14. Red: validation. Blue: test. Accuracy for different values of the regularization. (a) $\alpha^2 = 0.01$. (b) $\alpha^2 = 0.1$. (c) $\alpha^2 = 1$. The model configuration number in the horizontal axis correspond to the lexicographic order $1 = (1, 1, 1, 1, 1, 1)$, $2 = (1, 1, 1, 1, 1, 2)$, ...

Comparing these results of the restricted formulation (in Table V) with the unrestricted one (see Fig. 15), we observe significant differences in performance, which contrasts with all the previous scenarios. The performance of the restricted formulation is roughly $\sim 76\%$. In the unrestricted formulation, 11 configurations achieve test set accuracies above 80%. This suggests that, in this case, better discrimination is achieved by not sharing the motion fields, which is a natural conclusion, if one observed the typical trajectories in Fig. 12.

VII. COMPARISON WITH OTHER METHODS

In this section, we compare the proposed method to that of [63], using all the real datasets. We use a 5-fold cross-validation strategy, which is different from what was used in [63]. For a fair comparison, we run 8 times the EM algorithm with different initializations, and keep the best 5 estimates to compute an overall mean accuracy, for each of the experiments. The algorithm proposed in [63] has two stages. A first stage, where the low level parameters

TABLE VII

CLASSIFICATION RESULTS OF THE "MOVING LEFT" AND "MOVING RIGHT" CLASSES IN THE UCSD DATA USING [63]

	UCSD (Option i)	
	Down left	Up Right
Down Left	99.75%	0.25%
Up Right	1.11%	98.89%

	UCSD (Option ii)	
	Down left	Up right
Down Left	99.51%	0.49%
Up Right	0.74%	99.26%

	UCSD (Option iii)	
	Down left	Up Right
Down Left	99.01%	0.99%
Up Right	1.11%	98.89%

	UCSD (Option iv)	
	Down left	Up right
Down Left	99.51%	0.49%
Up Right	0.56%	99.44%

TABLE VIII

CLASSIFICATION RESULTS FOR THE SIX ACTIVITIES IN THE CAMPUS SCENARIO USING [63]: $a_1 \rightarrow$ ENTERING THE BUILDING, $a_2 \rightarrow$ LEAVING THE BUILDING, $a_3 \rightarrow$ WALKING PARALLEL, $a_4 \rightarrow$ WALK DIAGONALLY UP, $a_5 \rightarrow$ WALK DIAGONALLY DOWN, $a_6 \rightarrow$ WALK DIRECTLY INTO THE BUILDING

	CAMPUS (option i)					
	a_1	a_2	a_3	a_4	a_5	a_6
a_1	100%	0%	0%	0%	0%	0%
a_2	0%	98.18%	1.82%	0%	0%	0%
a_3	0%	2.86%	97.14%	0%	0%	0%
a_4	0%	0%	0%	100%	0%	0%
a_5	1.25%	0.63%	0.63%	0%	96.25%	1.25%
a_6	0%	0%	0%	0%	0%	100%

	CAMPUS (option ii)					
	a_1	a_2	a_3	a_4	a_5	a_6
a_1	96.67%	0%	0%	0%	0%	3.33%
a_2	0%	100%	0%	0%	0%	0%
a_3	0%	0%	100%	0%	0%	0%
a_4	1.54%	0%	0%	98.46%	0%	0%
a_5	0%	0%	0%	0%	100%	0%
a_6	0%	0%	0%	0%	0%	100%

	CAMPUS (option iii)					
	a_1	a_2	a_3	a_4	a_5	a_6
a_1	98.33%	0%	0%	0%	0%	1.67%
a_2	0%	100%	0%	0%	0%	0%
a_3	0%	0%	100%	0%	0%	0%
a_4	0%	0%	0%	100%	0%	0%
a_5	0%	0%	0%	0%	100%	0%
a_6	0%	0%	0%	0%	0%	100%

	CAMPUS (option iv)					
	a_1	a_2	a_3	a_4	a_5	a_6
a_1	100%	0%	0%	0%	0%	0%
a_2	0%	100%	0%	0%	0%	0%
a_3	0%	0%	100%	0%	0%	0%
a_4	0%	0%	0%	100%	0%	0%
a_5	0%	0%	0%	0%	100%	0%
a_6	0%	0%	0%	0%	0%	100%

TABLE IX

CLASSIFICATION RESULTS FOR THE CLIENT AND NONCLIENT ACTIVITIES IN THE SHOPPING CENTER USING [63]

		SHOPPING (option i)				
		Client		**Non Client**		
		E	L	P	IO1	IO2
Client	E	83.33%	0%	0%	0%	16.67%
	L	0%	84.45%	0%	6.67%	8.89%
Non Client	P	6.0%	0%	88.0%	0%	6.0%
	IO1	0%	7.69%	0%	56.92%	35.38%
	IO2	0%	0%	0%	7.5%	92.5%

		SHOPPING (option ii)				
		Client		**Non Client**		
		E	L	P	IO1	IO2
Client	E	63.33%	0%	0%	0%	36.67%
	L	2.22%	75.56%	0%	8.89%	13.33%
Non Client	P	4.0%	2.0%	90.0%	0%	4.0%
	IO1	1.54%	1.54%	0%	67.69%	29.23%
	IO2	0%	0%	0%	12.5%	87.5%

		SHOPPING (option iii)				
		Client		**Non Client**		
		E	L	P	IO1	IO2
Client	E	88.89%	0%	0%	0%	11.11%
	L	0%	96.30%	0%	0%	3.70%
Non Client	P	0%	0%	100%	0%	0%
	IO1	0%	1.28%	0%	70.51%	28.20%
	IO2	0%	0%	0%	18.75%	81.25%

		SHOPPING (option iv)				
		Client		**Non Client**		
		E	L	P	IO1	IO2
Client	E	73.33%	0%	0%	0%	26.67%
	L	0%	86.67%	0%	2.22%	11.11%
Non Client	P	6.0%	0%	94%	0%	0%
	IO1	0%	1.54%	0%	69.23%	29.23%
	IO2	0%	0%	0%	10.0%	90.0%

(mean translations and covariances) are estimated via EM, with a a built-in minimum message length (MML) model selection criterion (following [64]), disregarding activity classes. A second stage, in which the *high level* parameters (*i.e.* transition matrices) are estimated. Basically, this is achieved by fixing the first and second order statistics of all activities obtained in the first stage, and a per-class HMM (second run of the EM) is estimated where the transition matrix is individually obtained for each class. That algorithm allows controlling the covariance matrix as follows: (i) free covari-

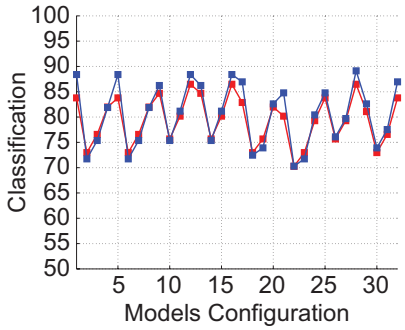

Fig. 15. Performance of the unrestricted formulation for the shopping scenario. Mean accuracy (over four initializations) on the: (red) five CV folds and (blue) test set. Notice that the validation generalizes quite well for the test set (the red curve follows the blue curve). As in Fig. 14, the model configuration number in the horizontal axis correspond to the lexicographic order $1 = (1, 1, 1, 1, 1, 1)$, $2 = (1, 1, 1, 1, 1, 2)$, ….

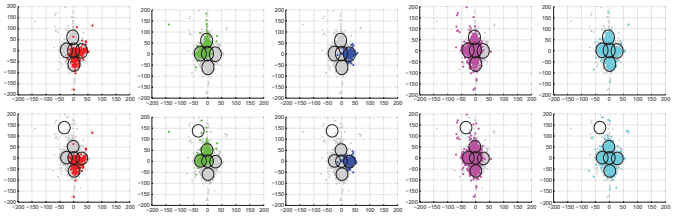

Fig. 16. EM mixture estimates with diagonal covariance matrices (option ii) in the shopping scenario. One cluster is spent to represent an outlier displacement.

ances; (ii) diagonal covariances, (iii) common covariance for all components and (iv) common diagonal covariance for all components (as in [64]). Comparing to that approach, two main difference characterize the present contribution: all the model parameters are jointly estimated by EM in a single stage and the transition matrices and the vector fields are both space-variant. In the experiments, we vary the number of models from 1 to 8 and run the EM algorithm 32 times (8 times for each covariance configuration). We compute mean accuracy over the best 5 estimates. Tables VII, VIII, IX (in the Appendix) show the performance of this algorithm in the UCSD, campus, and shopping scenarios, respectively, for all options of the covariance configurations. Table VI summarizes the results, where each cell contains the mean accuracy.

Concerning this last scenario (shopping), we observed strong discontinuities and gaps (pedestrian occlusions) in several trajectories. Since the algorithm operates on trajectory displacements, these discontinuities/gaps constitute outliers that may affect the mixture parameter estimates. Fig. 16 illustrates this issue, by showing how a single outlier can dominate one to the mixture components. To avoid this problem, we discard displacements above a given threshold. In the other real datasets (UCSD and campus) this pre-processing was not needed.

We conclude that the proposed approach exhibits a competitive performance with the method proposed in [63]. One advantage of the proposed framework is that no pre-processing is required in any of the real data sets, since the presented algorithm provides a regularization over the vector field estimates, giving more robustness with respect to trajectory discontinuities.

VIII. Conclusion

In this paper, we have presented a method for classification of trajectories, with application in video surveillance. The class-conditional generative models underlying the classifier is based on mixtures of vector fields with a location-dependent probabilistic switching mechanism.

We have proposed an EM algorithm to estimated the underlying motion fields along with the space-dependent switching probabilistic model. The estimates are based on finite-dimensional parameterizations of all the fields, based on which we place a smoothness-inducing Gaussian prior on the motion fields. Almost all the update equations of the EM algorithm have simple closed form expressions, with the exception of the update of the field of stochastic matrices. To solve this update equation, we have proposed a gradient projection algorithm, based on a state-of-the-art fast algorithm to compute the projection onto the probability simplex.

Furthermore, we presented two different schemes for selecting an appropriate model order. The first scheme uses shared motion fields, irrespective to the activity classes, that is, restricting the number of models within activities. The second scheme uses a discriminative model selection approach, which is able to automatically choose the number of underlying motion fields using a validation set.

Experiments using both synthetic and real data have shown that the proposed approach is able to model a wide range of motion patterns with an appropriate choice of the number of vector fields. The experiments have also testified for the the ability of the proposed EM algorithm to estimate the motion and switching fields from observed trajectories.

Appendix

This appendix provides the performance (Tables VII, VIII, and IX) of the methods described in Section VII for a detailed comparison regarding all the activities observed in all real data sets.

References

[1] D. Hu, S. Pan, V. Zheng, N. Liu, and Q. Yang, "Real world activity recognition with multiple goals," in *Proc. ACM 10th Int. Conf. Series*, 2008, pp. 30–39.

[2] M. Pantic, A. Pentland, A. Nijholt, and T. Huang, "Human computing and machine understanding of human behavior: A survey," in *Proc. 8th Int. Artif. Intell. Human Comput. Conf.*, 2007, pp. 239–248.

[3] A. Pentland, "Smart rooms: Machine understanding of human behavior," in *Proc. Comput. Vis. Human-Mach. Inter. Conf.*, 1998, pp. 1–353.

[4] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Syst. Cybern. C, Appl. Rev.*, vol. 34, no. 3, pp. 334–352, Aug. 2004.

[5] O. Javed and M. Shah, *Automated Multicamera Surveillance Algorithms and Practice*. New York: Springer-Verlag, 2008.

[6] J. Aggarwal and S. Park, "Human motion: Modeling and recognition of actions and interactions," in *Proc. 3-D Data Process., Vis. Trans. Conf.*, Sep. 2004, pp. 640–647.

[7] D. Gavrila, "The visual analysis of human movement: A survey," *Comput. Vis. Image Understand.*, vol. 73, no. 1, pp. 82–98, Jan. 1999.

[8] T. B. Moeslund, A. Hilton, and V. Kruger, "A survey of advances in vision-based human motion capture analysis," *Comp. Vis. Image Understand.*, vol. 104, pp. 90–126, Oct. 2006.

[9] N. Robertson and I. Reid, "A general method for human activity recognition in video," *Comput. Vis. Image Understand.*, vol. 104, no. 2, pp. 232–248, 2006.

[10] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Discriminative model selection for object motion recognition," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 3953–3956.

[11] M. Perse, M. Kristan, S. Kovacic, G. Vuckovic, and J. Persa, "A trajectory-based analysis of coordinated team activity in a basketball game," *Comput. Vis. Image Understand.*, vol. 113, no. 5, pp. 612–621, 2009.

[12] A. D. Wilson and A. F. Bobick, "Parametric hidden Markov models for gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 9, pp. 884–900, Sep. 1999.

[13] 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.

[14] V. Kellokumpu, G. Zhao, and M. Pietikäinen, "Human activity recognition using a dynamic texture based method," in *Proc. Brit. Mach. Vis. Conf.*, 2008, pp. 1–10.

[15] 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, Jun. 1997.

[16] H. Fujiyoshi, A. J. Lipton, and T. Kanade, "Real-time human motion analysis by image skeletonization," *IEICE Trans. Inf. Syst.*, vol. 87, no. 1, pp. 113–120, Jan. 2004.

[17] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Behavior classification by eigendecomposition of periodic motions," *Pattern Recognit.*, vol. 38, no. 7, pp. 1033–1043, 2005.

[18] R. Poppe, "Vision-based human motion analysis: An overview," *Comput. Vis. Image Understand.*, vol. 108, pp. 4–18, Jan. 2007.

[19] L. Rabiner, "A tutorial on hidden markov models and selected applications inspeech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[20] Y. Du, F. Chena, W. Xua, and W. Zhanga, "Activity recognition through multi-scale motion detail analysis," *Neurocomputing*, vol. 71, no. 18, pp. 3561–3574, 2008.

[21] T. Duong, H. Bui, D. Phung, and S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-Markov model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 838–845.

[22] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Independent increment processes for human motion recognition," *Int. J. Comput. Vis. Image Understand.*, vol. 109, no. 2, pp. 126–138, 2008.

[23] N. Oliver, B. Rosario, and A. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 831–843, Aug. 2000.

[24] Q. Shi and L. Wang, "Disciminative human action segmentation and recognition using semi-Markov model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jan. 2008, pp. 1–6.

[25] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based reprsentation," *IEEE Trans. Pattern Anal. Mach Intell.*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.

[26] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. 2nd Joint IEEE Int. Workshop, Vis. Surv. Perform. Eval. Track. Surv.*, Oct. 2005, pp. 65–72.

[27] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. 264–271.

[28] I. Laptev, "On space-time interest points," *Int. J. Comp. Vis.*, vol. 64, no. 3, pp. 107–123, 2005.

[29] F. Lv and R. Nevatia, "Single view human action recognition using key pose matching and viterbi path searching," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[30] H. Wang, M. Ullah, A. Kläser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2009, pp. 1–10.

[31] A. B. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 909–926, May 2008.

[32] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[33] O. Boiman and M. Irani, "Detecting irregularities in images and in video," in *Proc. Int. Conf. Comput. Vis.*, 2005, pp. 462–469.

[34] J. Liu and M. Shah, "Learning human actions via information maximization," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[35] P. Yan, S. M. Khan, and M. Shah, "Learning 4-D action feature models for arbitrary view action recognition," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–7.

[36] Z. Fu, W. Hu, and T. Tan, "Similarity based vehicle trajectory clustering and anomaly detection," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2005, pp. 602–606.

[37] N. Johnson and D. C. Hogg, "Learning the distribution of object trajectories for event recognition," *Image Vis. Comput.*, vol. 14, no. 8, pp. 583–592, 1996.

[38] X. Wang, K. Tieu, and E. Grimson, "Learning semantic scene models by trajectory analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 110–123.

[39] M. Pierobon, M. Marcon, A. Sarti, and S. Tubaro, "Clustering of human actions using invariant body shape descriptor and dynamic time warping," in *Proc. IEEE Conf. Adv. Video Signal Based Surv.*, Sep. 2005, pp. 22–27.

[40] M. Vlachos, G. Kollios, and D. Gunopulos, "Discovering similar multidimensional trajectories," in *Proc. Int. Conf. Data Eng.*, 2002, pp. 673–685.

[41] J. Berclaz, F. Fleuret, and P. Fua, "Multi-camera tracking and atypical motion detection with behavioral maps," in *Proc. 10th Eur. Conf. Comput. Vis. Conf.*, Oct. 2008, pp. 112–125.

[42] G. Heitz and D. Koller, "Learning spatial context: Using stuff to find things," in *Proc. 10th Eur. Conf. Comput. Vis.*, Oct. 2008, pp. 30–43.

[43] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–40.

[44] A. Torralba, "Contextual priming for object detection," *Int. J. Compon. Vis.*, vol. 53, no. 2, pp. 169–191, 2003.

[45] L. Li and L. F. Fei, "What, where and who? Classifying events by scene and object recognition," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[46] A. Gupta and L. Davis, "Objects in action: An approach for combining action understanding and object perception," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[47] M. Ryoo and J. Aggarwal, "Hierarchical recognition of human activities interacting with objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[48] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. Regh, "A scalable approach to activity recognition based on objects use," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[49] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *Proc. Conf. Comput. Vis. Pattern Recognit.* 2009, pp. 1–8.

[50] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[51] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from video in the wild," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1–8.

[52] H. Wang, A. Kläser, C. Schmid, and L. Cheng-Lin, "Action recognition by dense trajectories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Apr. 2011, pp. 3169–3176.

[53] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc., B*, vol. 39, no. 1, pp. 1–38, 1977.

[54] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[55] P. Dierckx, *Curve and Surface Fitting with Splines*. New York: Oxford Univ. Press, 1993.

[56] J. Nocedal and S. Wright, *Numerical Optimizatization*. New York: Springer-Verlag, 2006.

[57] J. Duchi, S. Shalev-Shwartz, and T. C. Y. Singer, "Efficient projections onto the ℓ_1-ball for learning in high dimensions," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 272–279.

[58] B. Thiesson and C. Meek, "Discriminative model selection for density models," in *Proc. 9th Int. Workshop Artif. Intell. Stat.*, 2003, pp. 1–6.

[59] A. B. Chan, Z. S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models for tracking," in *Proc. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–7.

[60] 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.

[61] J. Nascimento and J. S. Marques, "Performance evaluation of object detection algorithms for video surveillance," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 761–774, Apr. 2006.

[62] D. Huttenlocher and S. Ullman, "Recognizing solid objects by alignment with an image," *Int. J. Comput. Vis.*, vol. 5, no. 2, pp. 195–212, 1990.

[63] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Trajectory classification using switched dynamical hidden markov models," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1338–1348, Jun. 2010.

[64] M. A. T. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 381–396, Mar. 2002.

[65] A. Lanterman, *Schwarz, Wallace, and Rissanen: Intertwining Themes in Theories of Model Selection*. Univ. Illinois Press, Urbana, IL, 2000.

[66] D. Makris and T. Ellis, "Automatic learning of an activity based semantic scene model," in *Proc. Adv. Video Signal Based Surv.*, 2003, pp. 183–188.

[67] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Vector fields estimation for motion in natural images," in *Proc. IEEE Int. Conf. Imag. Process.*, Jun. 2009, pp. 1–10.

Jacinto C. Nascimento (S'00–M'06) received the E.E. degree from the Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the Instituto Superior Técnico (IST), Technical University of Lisbon, Lisbon, in 1995, 1998, and 2003, respectively.

He is currently a Principal Researcher of an FCT project with the Institute for Systems and Robotics, IST. He has authored or co-authored over 90 papers in international journals and conference proceedings (more in the IEEE). His current research interests include statistical image processing, pattern recognition, machine learning, medical imaging analysis, video surveillance, and general visual object classification.

Dr. Nascimento was on the program committees of many international conferences and is a Reviewer for several international journals.

Mário A. T. Figueiredo (S'87–M'95–SM'00–F'10) received the E.E., M.Sc., Ph.D., and "Agregado" degrees in electrical and computer engineering from the Instituto Superior Técnico (IST), Engineering School, Technical University of Lisbon, Lisbon, Portugal, in 1985, 1990, 1994, and 2004, respectively.

He has been with the Faculty of the Department of Electrical and Computer Engineering, IST, since 1994, where he is currently a Full Professor. He is a Group and Area Coordinator with the Instituto de Telecomunicações, Lisbon, a private not-for-profit research institution. He was with the Department of Computer Science and Engineering, Michigan State University, East Lansing, and the Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, in 1998 and 2005, respectively. His current research interests include signal processing and analysis, pattern recognition, machine learning, and optimization.

Dr. Figueiredo was the recipient of the 2011 IEEE SPS Best Paper Award, the 1995 Portuguese IBM Scientific Prize, and the 2008 UTL/Santander-Totta Scientific Prize. He is a fellow of the International Association for Pattern Recognition. From 2005 to 2010, he was a member of the Image, Video, and Multidimensional Signal Processing Technical Committee of the IEEE Signal Processing Society. He has been an Associate Editor of several journals, namely the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON MOBILE COMPUTING, *Pattern Recognition Letters,* and *Signal Processing*. He was the Co-Chair of the 2001 and 2003 Workshops on Energy Minimization Methods in Computer Vision and Pattern Recognition, a Guest Co-Editor of special issues of several journals, and a Program/Technical Committee Member of many international conferences.

Jorge S. Marques received the E.E. and Ph.D. degrees in electrical and computer engineering and the aggregation title from the Technical University of Lisbon, Lisbon, Portugal, in 1981, 1990, and 2002, respectively.

He is currently an Associate Professor with the Electrical and Computer Engineering Department, Instituto Superior Técnico, Lisbon, Portugal, and a Researcher with the Institute for Systems and Robotics. He has authored or co-authored over 150 papers in international journals and conferences, and has authored the book entitled *Pattern Recognition: Statistical and Neural Methods,* 2nd ed., in Portuguese (Lisbon, Portugal, IST Press, 2005). His current research interests include statistical image processing, shape analysis, and pattern recognition.

Dr. Marques was the Co-Chairman of the IAPR Conference IbPRIA 2005, the President of the Portuguese Association for Pattern Recognition from 2001 to 2003, and an Associate Editor of *Statistics and Computing Journal* (Springer).