Fast Estimation of Multiple Vector Fields: Application to Video Surveillance

Jorge S. Marques
Instituto de Sistemas e Robótica
Instituto Superior Tecnico
Lisboa, Portugal

Mário A. T. Figueiredo
Instituto de Telecomunicações
Instituto Superior Tecnico
Lisboa, Portugal

***Abstract*—Characterizing object motion in a given scene is a central problem in computer vision and image analysis. Object motion has been recently modeled by using multiple motion fields; this model allows characterizing typical motion patterns and, among other possible applications, may be used to detect abnormal events. However, the estimation of multiple fields from video information (e.g., trajectories) is a challenging task since we do not know which field is active at each instant of time, for each object. This difficulty has been successfully addressed by using iterative approaches in which the estimation of the active field alternates with the field update, using the expectation-maximization (EM) algorithm or variants thereof. However, the EM method for this problem has been shown to be slow and to yield field estimates that depend on the initialization. This paper describes an alternative approach for the estimation of multiple overlapping fields, using a label propagation algorithm. The proposed algorithm, which is not iterative, is fast and has good performance on synthetic and real data.**

I. Introduction

Most video surveillance systems involve tracking objects of interest (usually pedestrians or vehicles) in the scene, in order to characterize their activities and detect abnormal behaviors [15]. Several cues are used to accomplish this goal, such as shape and motion features, or even articulated models of the human body [12], [13]. These techniques are applicable when the camera field of view is small and the objects of interest are sufficiently close to the camera; in contrast, when the surveillance camera(s) covers a wide area, it is no longer possible to reliably extract the detailed information required by those methods. In these scenarios, the most reliable information is carried by the trajectories of the centers of mass of the objects in the scene.

Several methods have been developed in the last decade to compare trajectories and to cluster them [8], [9]. Those methods allow the system to characterize typical behaviors and detect unusual ones, a task for which several approaches have been followed. Some authors compare trajectories using dissimilarity measures, such as the Euclidean or the Hausdorff distances [2], [6]; since the observed trajectories in general have different lengths, these approaches require some kind of alignment of the trajectories. This is often done using dynamic programming methods, such as dynamic time warping [7]. The trajectory alignment compensates for small deformations but it performs poorly in the presence of deformations changes.

This work was supported by FCT (plurianual funding) through the PIDDAC Program funds and by project PTDC/EEACRO/098550/2008.

Fig. 1. Main goal: learn motion patterns

To avoid the need for alignment, some authors have adopted generative models. These include *hidden Markov models* (HMM) [5], probabilistic landmarks [11], and the recently proposed *mixtures of motion fields* (MMF) [10]. The MMF model is based on two key observations: motion fields are intuitive (they have a clear physical meaning) and flexible representations; however, a single motion field is usually not enough to model a set of diverse or complex trajectories, possibly exhibiting intersections (as illustrated in Fig. 1). For example, if the trajectory is generated by a single differential equation $\dot{x}(t) = T(x(t))$, where $x(t)$ denotes the position of the object at time t and $T(.)$ denotes the motion field, then trajectory intersections would violate the uniqueness theorem for differential equations. The use of a mixture of multiple motion fields is therefore required and field switching has to be considered.

The joint estimation of multiple motion fields and of the switching probabilities was addressed using an EM iterative algorithm [10]. That approach is time consuming and the field estimates obtained by the EM method depend on the initialization. This paper proposes an alternative approach by addressing the following question: *can we estimate multiple vector fields in a fast, non-iterative way?* The algorithm should account for multiple motion fields, with a non-uniform structure in space, and should be able to automatically select the number of fields from the data. Furthermore, it should avoid an iterative refinement of the solution. The algorithm

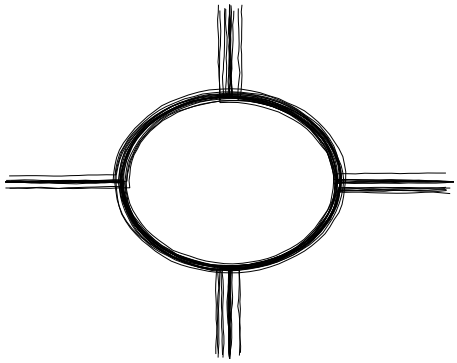

Fig. 2. Roundabout example: object trajectories

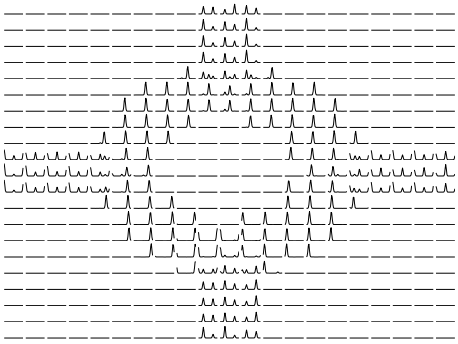

Fig. 3. Roundabout example: local histograms

proposed in this paper meets all these desiderata: it is a non-iterative algorithm, based on local decisions and local statistics of the data, and it is based on label propagation.

II. Overview

Our approach is supported on the two following main assumptions: (i) we have a *large amount of data*; (ii) there are *typical motion regimes* at each point in the scene, *i.e.*, the velocity is a random vector with a small number of modes. The first assumption is valid in most applications, since we can easily track dozens or even hundreds of objects in the scene. On the contrary, the second assumption is not always true, as it depends on the underlying scenario. For example, consider the trajectories of pedestrians in a park: we may observe a wide variety of trajectories and the uncertainty may be high. However, in other more structured scenarios (*e.g.*, streets, train stations, university campi), people tend to traverse the space in typical paths (*e.g.*, they walk along streets, they cross streets, they enter and leave buildings), thus we may hope to learn these typical paths automatically from data. The same applies if we are surveying vehicles in structured environments (such as roads, streets, or parking lots).

Assuming the availability of a large number of observed trajectories, we can characterize the velocity vector at each point in the image domain by a velocity histogram. Furthermore, the histogram provides a hint about the number of fields required to describe the typical motions at that location. We will assume that if an histogram has m modes, then there are m active motion fields at that location in space. Correspondingly, we will assume that each peak of the histogram corresponds to one of the underlying motion fields. Figures 2 and 3 illustrate these ideas with a synthetic example; Figure 2 shows a set of trajectories and Figure 3 shows histograms of the velocity direction computed from the trajectories. These are local histograms computed at the nodes of a uniform 21×21 grid and using local information (trajectories) in the vicinity of each node. Many histograms in this example exhibit two peaks separated by π, corresponding to motion trajectories in opposite directions. Other histograms have a larger number of peaks, such as those at the intersection of the straight and circular trajectories.

The main question is: *how can we associate the peaks to motion fields and label them in a consistent way, in order to obtain smooth vector fields?*

To answer the question just formulated, we will adopt a region growing approach. We will choose a grid node as a seed and initialize the first motion field at this point with the direction associated to the largest histogram peak. A new label is created at this point. Then, we propagate the vector field (label) to the neighboring nodes, provided that they meet the following conditions:

- the histogram of the neighboring cell has a peak close to the motion estimate,
- the peak was not previously assigned to any motion field (unlabeled peak).

This corresponds to a non-iterative label propagation algorithm in which each data location is visited only once, thus it is much faster then the EM method. In the next sections, we describe in detail the histogram computation and the label propagation algorithm.

III. Local motion models

Consider the trajectories (extracted from a video sequence) of S objects (*e.g.*, pedestrians, vehicles), $\mathcal{X} = \left(x^{(1)}, ..., x^{(S)}\right)$, where the trajectory of the k-th object, $x^{(k)}$, is a sequence of L_k positions on the plane, $x^{(k)} = \left(x_1^{(k)}, \ldots, x_{L_k}^{(k)}\right)$, with

$$x_t^{(k)} = \begin{bmatrix} \left(x_t^{(k)}\right)_1 \\ \left(x_t^{(k)}\right)_2 \end{bmatrix} \in \mathbb{R}^2.$$

Typically, this amounts to thousands (or even tens or hundreds of thousands) of data vectors, and our goal is to statistically characterize this information, using a set of local histograms, computed at the nodes of a uniform grid, as illustrated in Figure 3.

Assuming uniform sampling with unit time steps (without any loss of generality), let $v_t^{(k)} = x_{t+1}^{(k)} - x_t^{(k)}$ be the velocity vector at time t for object k. We will assume that the velocity direction,

$$\theta_t^{(k)} = \arctan \frac{\left(v_t^{(k)}\right)_2}{\left(v_t^{(k)}\right)_1},$$

is the most discriminative feature (Matlab function atan2 is used to obtain $\theta_t^{(k)}$ in the range $[-\pi, \pi[$). Thus we will

characterize motion by a set of local models: the histograms of θ values in the neighborhood of the image nodes u_i.

Considering a dataset of trajectories, $\mathcal{X} = \left(x^{(1)}, ..., x^{(S)}\right)$, the local (smoothed) histograms are obtained using a a Gaussian kernel [16] in the spatial dimension and a von Mises kernel [14] in the angular dimension. The von Mises density is the analog of the Gaussian density, for random variables defined on the circle, taking into account its 2π-periodic nature of θ (phase wrapping). The value of the local histogram at node u_i and direction bin θ_j is thus given by

$$h(u_i, \theta_j) = \gamma \sum_{k=1}^{S} \sum_{t=1}^{L_k - 1} w_t^{(k)}(u_i, \theta_j)\,, \tag{1}$$

where γ is a normalization factor and

$$w_t^{(k)}(u_i, \theta_j) = \mathcal{N}(x_t^{(k)} | u_i, \sigma^2 I)\, \mathcal{M}(\theta_t^{(k)} | \theta_j, \kappa) \tag{2}$$

is a weighting function, where $\mathcal{N}(\cdot|\mu, R)$ denotes the normal density function with mean vector μ and covariance matrix R, and

$$\mathcal{M}(\theta|\eta, \kappa) = \frac{e^{\kappa \cos(\theta - \eta)}}{2\,\pi\, I_0(\kappa)} \tag{3}$$

is a von Mises density of mean η and concentration parameter κ (with I_0 denoting the modified Bessel function of zero-th order). The local histograms for the example of Figure 2 are shown in Figure 3. In this example, the image domain $[0, 1]^2$ was covered by a regular grid with 21×21 nodes and the direction range $[-\pi, \pi[$ was quantized into 64 equal bins.

Of course we can also associate a velocity estimate to each histogram position and orientation (u_i, θ_j). This can be done by averaging all the displacements $v_t^{(k)}$ weighted by $w_t^{(k)}(u_i, \theta_j)$.

$$v(u_i, \theta_j) = \frac{\sum_{k=1}^{S} \sum_{t=1}^{L_k - 1} w_t^{(k)}(u_i, \alpha_j)\; v_t^{(k)}}{\sum_{k=1}^{S} \sum_{t=1}^{L_k - 1} w_t^{(k)}(u_i, \alpha_j)}\,. \tag{4}$$

The weights take into account the distance of $x_t^{(k)}$ to the node u_i and motion direction.

IV. Multiple Motion Fields Estimation

Let us consider Figure 3 again. When we have multiple motion fields in the vicinity of a node u_i, the corresponding histogram presents multiple peaks. The number of peaks is therefore an estimate of the number of motion fields that are active at a given location (the typical motion directions at that location). Let Ω_i be the set of directions associated to the local maxima of the histogram $h(u_i, \theta)$, as a function of θ. We now have one challenging problem: we would like to estimate a set of K motion fields from this information. This is an unsupervised labeling problem, since we do not have pre-defined classes associated to the labels. We would like to assign a label $l \in \{1, \ldots, K\}$ (with K itself unknown) to each histogram peak $\alpha \in \Omega_i$. A criterion must be defined to assign histogram peaks to different vector fields (labeling). In this paper we will assume that each vector field is smooth and direction changes in pedestrian trajectories correspond to switching between motion fields.

The labeling problem described in the previous paragraphs can be formulated as the minimization of an objective function. Let us denote by (α_i, u_i, l_i), for $i = 1, \ldots, P$, the sequence of all the histogram peaks, the corresponding nodes, and assigned labels. We can define a cost function which depends on all the absolute differences between peaks associated to neighboring nodes with the same label,

$$\begin{aligned} C(l_1, ..., l_P) &= \sum_{\substack{i,\,j\,:\,l_i = l_j \\ u_j \,\in\, N(u_i)}} \min_{k \in \mathbb{Z}} |\alpha_i - \alpha_j + 2\,k\,\pi| \\ &= \sum_{i \sim j} \min_{k \in \mathbb{Z}} |\alpha_i - \alpha_j + 2\,k\,\pi|\; \mathbb{I}(l_i = l_j) \end{aligned} \tag{5}$$

where $N(u_i)$ is the set of nodes which are neighbors of u_i according to some neighborhood criterion (e.g., 4-connected), $i \sim j$ is a shorthand for $u_i \in N(u_j)$ (which, of course, implies that $u_j \in N(u_i)$, because neighborhood relations are symmetric), the $2\,k\,\pi$ term accounts for the circular nature of angular differences (*e.g.*, for a very small ε, any angle of the form $2\,k\,\pi \pm \varepsilon$ is very close to 0), and $\mathbb{I}$ denotes the indicator function, that is $\mathbb{I}(A) = 1$, if A is true, and $\mathbb{I}(A) = 0$, if A is false.

Clearly, the maximum number of fields should also be specified; otherwise, we could trivially minimize C by simply increasing the number of fields.

The minimization of (5), with respect to the set of labels $l_i,\ i = 1, ..., P$, is a hard combinatorial problem, which could be addressed using classical methods, such as relaxation labeling [4] or simulated annealing with Gibbs sampling [3], or with newer tools such as graph cuts [1]. All these approaches are suboptimal and require predefinition of C.

Since we wish to obtain a non-iterative solution for the problem, we will adopt a label propagation approach, which can also be interpreted as a message passing algorithm. The proposed algorithm works as follows. First, we select one node and one peak of the histogram at that node, (u, α), as a seed, and assign a label to it. This initializes a set R of pairs (u, α) such that the nodes form a connected region and the peaks share the same label. Then, we propagate the labels of the boundary nodes to their neighbors, provided they meet three conditions next described. Let (u, α) be a member of R and (w, β) be a candidate pair consisting of a grid node, such that $w \in N(u)$, and an histogram peak. The label of (u, α) is propagated to (w, β) if the following conditions are met:

- $\min_k |\alpha - \beta + 2\pi k| < T$;
- (w, β) had not been previously given any label.

The region R grows until there are no more nodes and peaks meeting these conditions. When that happens, another seed is selected among the still unlabeled pairs, and the growing process is repeated. The algorithms stops when all the peaks are labeled.

The output of the algorithm depends, of course, on the initialization of the regions (the seeds), on the order used to

test the neighboring nodes and peaks, and on the value of the threshold T. In this paper we adopt a simple approach: as seeds, we select the nodes with lowest entropy histograms and the peaks with highest value. In the growing process, the neighboring nodes are visited in a random order. Other strategies could be chosen, such as a greedy strategy in which we would choose the neighbor node and peak (w, β) with smallest cost according to (5).

The next section presents some experimental results.

V. Results

The proposed algorithm was applied to estimate multiple motion fields from real and synthetic data. In the first case, the object trajectories were extracted from video sequences, while in the second case they were generated by a stochastic model. The trajectories were normalized to fit inside the unit square $[0, 1]^2$ and the motion fields are defined on a regular grid of 21×21 nodes. The orientation of the velocity vector was quantized into 64 bins and the kernel parameters were chosen as follows: $\sigma = 1/42$ and $\kappa = 128$. The maximum peak deviation was $T = 2$ in synthetic examples and $T = 1$ in the case of real data.

The algorithm was programmed in MATLAB without any special care to speed optimization. We have used two lists of nodes, termed "open" and "closed". The open list stores the indices of the nodes which meet the smoothness conditions and that should be labeled in future. The closed list includes the nodes which were already tested and labeled.

A. Synthetic data

Two synthetic experiments will be described. The first experiment uses trajectories generated by two motions regimes (circular and linear) without any switching between these motion regimes. It should be stressed that there is spatial overlap between both types of trajectories. Fig 4 shows the trajectories and local histograms as well as the estimated fields. The algorithm manages to separate both fields well, despite their spatial overlap.

The second synthetic experiment simulates the motion of vehicles in roundabout. Each vehicle enters in the roundabout through one of the four entries and may leave in any of the four exits with a given probability. The trajectories for this example were shown before (Figure 2) and the field estimates are displayed in Figure 5. The algorithm found 9 motion fields. A post-processing step automatically merged non-overlapping fields with the same direction, yielding the 5 fields shown in Figure 5. We obtained 1 circular field, and 4 linear (entering/leaving) fields, associated to the 4 entries/exits.

We conclude from these examples that the label propagation algorithm is able to determine the number of fields and to estimate overlapping motion fields generated by simple synthetic models. The next section considers a more challenging problem with real data.

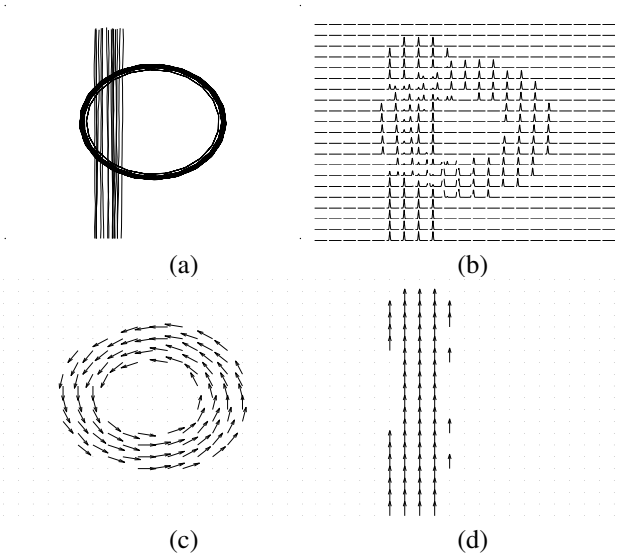

Fig. 4. Two overlapping fields: (a) trajectories, (b) local histograms, (c,d) estimated fields

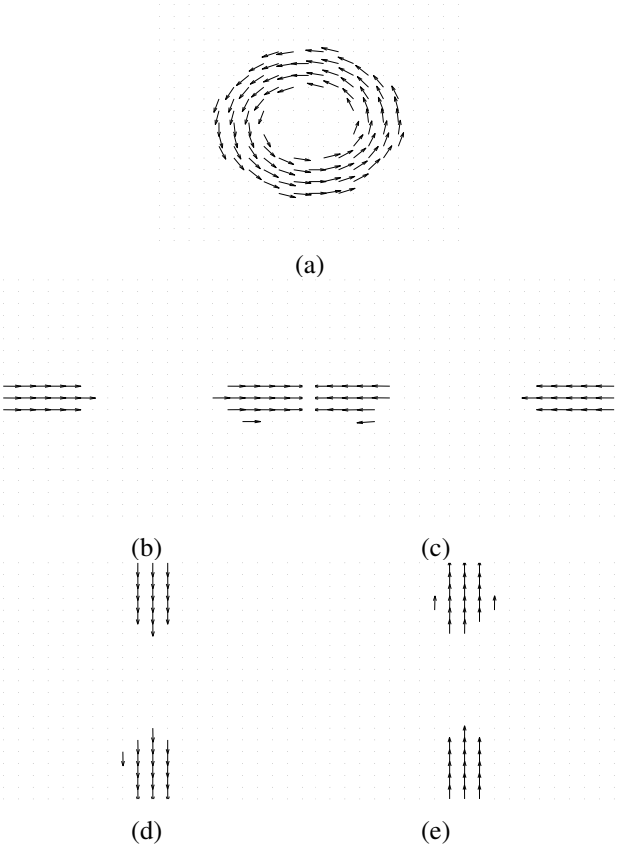

Fig. 5. The synthetic roundabout experiment: estimated fields.

B. University campus

This example was used as a benchmark in other works with multiple vector fields [10]. The data consists of 144 pedestrian trajectories (17284 points) obtained with a surveillance camera at the campus of a university (Figure 6(a)). The images were geometrically transformed to compensate for the perspective

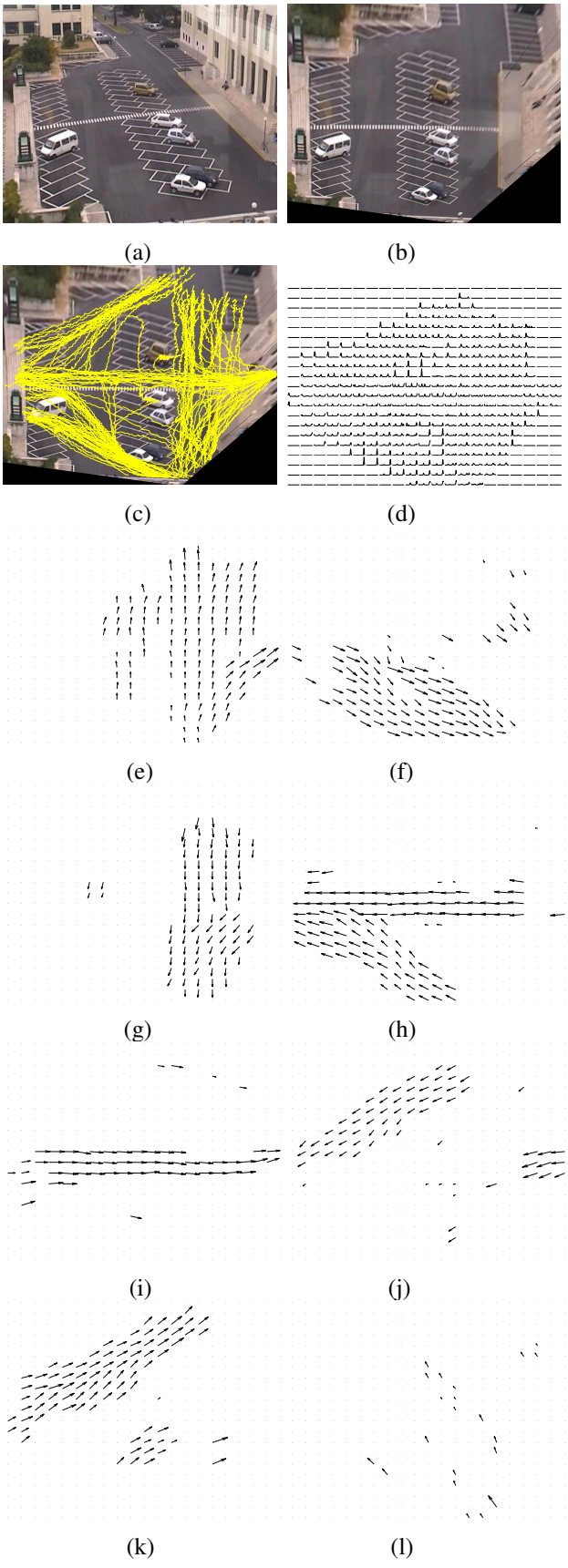

Fig. 6. campus example: (a) original image, (b) bird eye view, (c) object trajectories, (d) local histograms, (e-l) estimated fields

projection distortion using an homography (Figure 6(b)). The trajectories and corresponding histograms are shown in Figures 6(c) and 6(d).

The labeling algorithm generates a large number of fields, but most of them have a very small support (less than five nodes). The estimated fields were ordered according to their support area and Figure 6 shows the first 8, which represent typical motion patterns in this campus. Although these fields seem to match the data well, an objective quantitative evaluation of the model performance should be done. This issue is addressed in the next section.

C. Quantitative evaluation

To assess the performance of the proposed algorithm, we measure the ability of the estimated model to predict the object position one step ahead. There is one difficulty in this approach: we do not know which field is active at each instant of time. This difficulty is sidestepped by choosing the field that leads to the smallest prediction error.

We defined a prediction signal-to-noise-ratio (SNR, in dB) as follows

$$SNR(dB) = 10 \log_{10} \frac{E_v}{E_r} , \tag{6}$$

where E_v is the energy of the observed motion vectors

$$E_v = \sum_{k=1}^{S} \sum_{t=2}^{L_k} \|x_t^{(k)} - x_{t-1}^{(k)}\|^2 , \tag{7}$$

and E_r is the energy of the (minimum) prediction residue

$$E_r = \sum_{k=1}^{S} \sum_{t=2}^{L_k} \min_p (\epsilon_t^{(k)})^2 , \tag{8}$$

$$\epsilon_t^{(k)} = \|x_t^{(k)} - x_{t-1}^{(k)} - T_p(x_{t-1}^{(k)})\| \tag{9}$$

where T_p denotes the $p-$th vector field obtained by interpolating the motion vectors defined at the grid nodes using splines. The interpolation details can be found in [10]. The SNR measure is simple and takes into account the use of multiple fields.

The prediction errors for the campus example are shown in Figure 7. Each target position has a color. Green corresponds to very small residues ($\epsilon_t^{(k)} < 0.001$), yellow to middle size residues ($0.001 \leq \epsilon_t^{(k)} < 0.005$) and red to large residues ($\epsilon_t^{(k)} \geq 0.005$). The image domain was normalized to fit the interval $[0, 1]^2$. We conclude from this example that very good prediction is achieved in most of the observed positions with a small amount of large errors. It is also interesting to observe the prediction gain, defined by

$$G_t^{(k)} = \frac{\|x_t^{(k)} - x_{t-1}^{(k)}\|^2}{\min_p (\epsilon_t^{(k)})^2} ; \tag{10}$$

Figure 8 shows the prediction gain for each target position in the data set. We use a color code as before to visualize the gain magnitude. We apply a green label if the prediction error meets the condition $G_t^{(k)} > 10$, a yellow label if the prediction gain

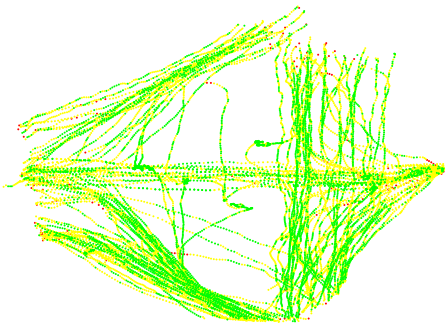

Fig. 7. Distribution of prediction error (color code: green if $\epsilon_t^{(k)} < 0.001$, yellow if $0.001 \leq \epsilon_t^{(k)} < 0.005$, red if $\epsilon_t^{(k)} \geq 0.005$)

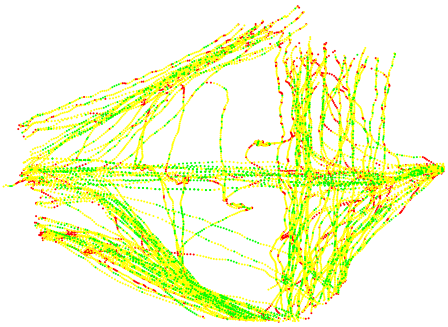

Fig. 8. Distribution of prediction gain (color code: green if $G_t^{(k)} > 10^2$, yellow if $2 \leq G_t^{(k)} < 10$, red if $G_t^{(k)} \leq 2$)

lies in the interval $2 \geq G_t^{(k)} > 10$, and red label if $G_t^{(k)} \leq 2$. Most of the positions in the data set receive green or yellow labels. Isolated trajectories which do not follow typical trends have worse prediction gains as expected.

The SNR results for the three examples are shown in table I. We consider two algorithms: the fast algorithm (termed *fast multiple vector fields estimation* – FMVFE) and the EM method. In the EM tests we used 20 iterations and 2, 5 and 8 fields respectively defined by the user while the fast algorithm estimates the number of fields in an automatic way. The proposed algorithm achieves very good scores and performs well even in the challenging example of the university campus. However, the best results are achieved with the EM method through a slow recursive fine tuning of the fields estimates.

It should be stressed that the proposed algorithm is much faster than the EM method as shown in table II. The FMVT time does not include the preprocessing step (histogram computation) which takes $9.34, 11.29, 90.62$ s in these examples. However this step can easily be done in real time.

TABLE I
EVALUATION OF THE FAST MULTIPLE VECTOR FIELDS (FMVF) ALGORITHM AND EXPECTATION-MAXIMIZATION (EM) ESTIMATION ACCORDING TO SNR (DB).

SNR (dB)	FMVF	EM
2 fields	31.4	36.1
roundabout	22.7	23.7
campus	10.9	15.1

TABLE II
COMPUTATION TIME OF THE FAST MULTIPLE VECTOR FIELDS (FMVF) ALGORITHM AND EXPECTATION-MAXIMIZATION (EM) METHOD.

CPU (sec)	FMVF	EM
2 fields	0.05	100.29
roundabout	0.09	569.75
campus	1.96	5984.25

VI. CONCLUSIONS

This paper presented a fast algorithm for the estimation of multiple velocity fields from object trajectories in the image. The algorithm is based on a local characterization of the velocity vector at a set of nodes, followed by a label propagation process which enforces global coherence within each field. The algorithm is fast, non-iterative (each site is visited only once) and deals with multiple overlapping fields. Experimental results show that good performances are achieved in synthetic and real examples.

Acknowledgement: The campus sequences were kindly provided by Dr. Jacinto Nascimento.

REFERENCES

[1] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimisation via graph cuts", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1222-1239, 2001.

[2] Z. Fu, W. Hu, and T. Tan, "Similarity based vehicle trajectory clustering and anomaly detection", *Proc. of the IEEE Internat. Conf. on Image Processing – ICIP*, vol. II, pp. 602–605, 2005.

[3] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741, 1984.

[4] D. Hummel and S.Zucker, "On the foundations of relaxation labelling processes", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 267–287, 1983.

[5] F. Jiang, Y. Wu, A. Katsaggelos, "Abnormal event detection from surveillance video by dynamic hierarchical clustering", *Proc. of the IEEE Internat. Conf. on Image Processing*, vol. V, pp. 145–148, 2007.

[6] I. Junejo, O. Javed, and M. Shah, "Multi feature path modeling for video surveillance," *Proc. of International Conference on Pattern Recognition*, vol. 2, pp. 716–719, 2004.

[7] E. J. Keogh and M. J. Pazzani, "Scaling up dynamic time warping for data mining application", *Proc. of the Internat. Conf. on Knowledge Discovery and Data Mining*, pp. 285-289, 2000.

[8] J. Melo, A. Naftel, A. Bernardino, J. Santos-Victor, "Detection and classification of highway lanes using vehicle motion trajectories", *IEEE Trans. on Itelligent Transportation Systems*, vol. 7, pp. 188–200, 2006.

[9] B. T. Morris and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance", *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, pp. 1114–1127, 2008.

[10] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques, "Trajectory analysis in natural images using mixtures of vector fields", *Proc. of the IEEE Internat. Conf. on Image Processing – ICIP*, 4353–4356, 2009.

[11] S. Parise and P. Smyth, "Learning stochastic path planning models from video images", technical report 04–12, School of Information and Computer Science, University of California at Irvine, 2004.

[12] S. Park, J. K. Aggarwal, "Simultaneous tracking of multiple body parts of interacting persons", *Computer Vision and Image Understanding*, vol. 102, pp. 1–21, 2006.

[13] D. Ramanan, D. A. Forsyth, A. Zisserman, "Tracking people by learning their appearance", *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 29, pp. 65–81, 2007.

[14] S. Rao Jammalamadaka and A. Sengupta, *Topics in Circular Statistics*, World Scientific Publishing, 2001.

[15] C. Regazzoni, V. Ramesh, G. Foresti (editors), *Special issue on video communications, processing, and understanding for third generation surveillance systems, Proceedings of the IEEE*, vol. 89, no. 10, 2001.

[16] D. W. Scott, *Mulivariate Density Estimation*, Wiley-Interscience, 1992.