

# Efficient Optimization Algorithm for Space-Variant Mixture of Vector Fields<sup>\*</sup>

Jacinto C. Nascimento<sup>1,2</sup>, Miguel Barão<sup>3,4</sup>, Jorge S. Marques<sup>1,2</sup>, and João M. Lemos<sup>1,3</sup>

<sup>1</sup> Instituto Superior Técnico, Portugal

<sup>2</sup> Instituto de Sistemas e Robótica, Portugal

<sup>3</sup> INESC-ID, Portugal

<sup>4</sup> Universidade de Évora, Portugal

**Abstract.** This paper presents a new algorithm for trajectory classification of human activities. The presented framework uses a mixture of parametric space-variant vector fields to describe pedestrian’s trajectories. An advantage of the proposed method is that the vector fields are not constant and depend on the pedestrian’s localization. This means that the *switching motion* among vector fields may occur at any image location and should be accurately estimated. In this paper, the model is equipped with a novel methodology to estimate the switching probabilities among motion regimes. More specifically, we propose an iterative optimization of switching probabilities based on the natural gradient vector, with respect to the Fisher information metric. This approach follows an information geometric framework and contrasts with more traditional approaches of constrained optimization in which euclidean gradient based methods are used combined with probability simplex constraints. We testify the performance superiority of the proposed approach in the classification of pedestrian’s trajectories in synthetic and real data sets concerning far-field surveillance scenarios.

**Keywords:** EM, natural gradient, information geometry, surveillance, trajectories, vector fields.

## 1 Introduction

The plethora of related work concerning human activity recognition is significantly large and often divided into several categories depending on the setup used. This usually comprises, single-view, single-view-invariant and multi-view settings which are amongst the most used systems for surveillance tasks. In this work, a far-field single view fixed camera is adopted. This allows to obtain the same camera viewpoint that is useful to collect the trajectories unfolded in the scenario.

---

<sup>\*</sup> This work was supported by the FCT project [PEst-OE/EEI/LA0009/2011] and Project ARGUS (PTDC/EEA-CRO/098550/2008).

The framework proposed herein, uses a parametric model to describe the motion vector fields as in [1] with a new efficient optimization algorithm for the switching among these fields inspired in [2]. Therefore, the proposed approach is twofold. First, the vector fields are not constant, thus providing a flexible model and expressive representation of complex motion patterns in the pedestrian’s trajectories. Second, the optimization algorithm relies on the use of the natural gradient with respect to the Fisher information metric to model the switching probabilities among the vector fields. This technique follows an information geometric framework and has the following main properties: (i) the use of the Fisher metric modifies the euclidean gradient direction so that probabilities are constrained to the probability simplex, and (ii) the natural gradient can, in some cases, be viewed as a quasi-Newton method, since the Fisher information approaches asymptotically the Hessian of several common information divergences. These two properties suggest faster convergence rates and reduced computational effort contrasting with the gradient projection based methods.

## 2 Generative motion model - Multiple vector fields

We will denote the set of vector motion fields as  $\mathcal{T} = \{\mathbf{T}_1, \dots, \mathbf{T}_K\}$ , with  $\mathbf{T}_{k_t} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ , for  $k_t \in \{1, \dots, K\}$ . The generative motion model of the trajectory is given as

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

where  $\mathbf{w}_t \sim \mathcal{N}(0, \sigma_{k_t}^2 \mathbf{I})$  is white Gaussian noise with zero mean and variance  $\sigma_{k_t}^2$  (which may be different for each field), and  $L$  is the number of points in the trajectory. Also, we will assume that the sequence of active fields  $\mathbf{k} = \{k_1, \dots, k_L\}$  is modeled as a realization of a first order Markov process with space varying transition probabilities. This model allows the switching to depend on the object localization, thus having  $P(k_t = j | k_{t-1} = i, \mathbf{x}_{t-1}) = \mathbf{B}_{ij}(\mathbf{x}_{t-1})$ , where  $\mathbf{B} : \mathbb{R}^2 \rightarrow \mathbb{R}^{K \times K}$  is a field of stochastic matrices. The matrix  $\mathbf{B}$  can also be seen as a set of  $K^2$ -dimensional fields with values in  $[0, 1]$  s.t.  $\sum_j B_{ij}(\mathbf{x}_t) = 1$ , for any  $\mathbf{x}_t$  and any  $i$ .

The joint distribution of a trajectory  $\mathbf{x}$  and its underlying sequence of active fields  $\mathbf{k}$ , under the model parameters  $\Theta = (\mathcal{T}, \mathbf{B}, \Sigma)$ , is given by

$$p(\mathbf{x}, \mathbf{k} | \Theta) = p(\mathbf{x}_1) P(k_1) \prod_{t=2}^L p(\mathbf{x}_t | \mathbf{x}_{t-1}, k_t) p(k_t | k_{t-1}, \mathbf{x}_{t-1}). \quad (2)$$

From (2), we see that  $p(k_t | k_{t-1}, \mathbf{x}_{t-1})$  is a function of  $\mathbf{B}$ ,  $p(\mathbf{x}_t | \mathbf{x}_{t-1}, k_t)$  is a function of  $\mathcal{T}$  and  $\Sigma$ , and  $p(\mathbf{x}_t, k_t | \mathbf{x}_{t-1}, k_{t-1})$  is a function of  $\mathcal{T}$ ,  $\mathbf{B}$ , and  $\Sigma$ .

As in [1] both of the fields and transition matrices ( $\mathcal{T}, \mathbf{B}$ ) are modeled in a parametric way. More specifically, they are defined at the nodes of a regular grid. To obtain the velocity fields and switching probability fields, we interpolate the

vectors  $\mathbf{t}_{\mathbf{k}}^{(n)}$  and matrices  $\mathbf{b}^{(n)}$  defined at the nodes of the grid as follows

$$\mathbf{T}_k(\mathbf{x}) = \sum_{n=1}^N \mathbf{t}_k^{(n)} \phi_n(\mathbf{x}), \quad \mathbf{B}(\mathbf{x}) = \sum_{n=1}^N \mathbf{b}^{(n)} \phi_n(\mathbf{x}) \quad (3)$$

where  $\phi_n(\mathbf{x}) : \mathbb{R}^2 \rightarrow \mathbb{R}$ , for  $n = 1, \dots, N$  is a set of scalar basis functions. Given the image domain  $\mathcal{D} = [0, 1]^2$ , a discretization is performed using an uniform grid with step  $\Delta$ . The contribution of node  $n$ , centered at  $\mathbf{u}_n = (u_n^1, u_n^2)$ , to the interpolation (3) is given by

$$\phi_n(\mathbf{x}) = \begin{cases} |x^1 - u_n^1| \cdot |x^2 - u_n^2| / \Delta^2 & \text{if } |x^1 - u_n^1| < \Delta \text{ and } |x^2 - u_n^2| < \Delta, \\ 0 & \text{otherwise.} \end{cases}$$

### 3 Learning the model

Here, we detail how the model parameters  $\Theta = (\mathcal{T}, \mathbf{B}, \Sigma)$  are learned. More specifically, how the motion fields  $\mathcal{T} = \{\mathbf{T}_1, \dots, \mathbf{T}_K\}$ , the field of the stochastic matrices  $\mathbf{B}$  and the noise variances  $\Sigma = \{\sigma_1^2, \dots, \sigma_K^2\}$  are learned from a set of  $S$  independent observed trajectories  $\mathcal{X} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(S)}\}$ , where  $\mathbf{x}^{(s)} = (\mathbf{x}_1^{(s)}, \dots, \mathbf{x}_{L_s}^{(s)})$  is the  $s$ -th observed trajectory. Since we assume that the active models  $\mathcal{K} = \{\mathbf{k}^{(1)}, \dots, \mathbf{k}^{(S)}\}$  are missing, we apply the EM algorithm to find a *marginal maximum a posteriori* (MMAP) estimate of  $\Theta$ ; formally the estimate is given by

$$\begin{aligned} \hat{\Theta} &= \arg \max_{\Theta} \sum_{\mathcal{K}} p(\mathcal{X}, \mathcal{K} | \Theta) p(\Theta) \\ &= \arg \max_{\Theta} \sum_{\mathcal{K}} \prod_{s=1}^S p(\mathbf{x}^{(s)}, \mathbf{k}^{(s)} | \Theta) p(\Theta) \end{aligned} \quad (4)$$

where each factor  $p(\mathbf{x}^{(s)}, \mathbf{k}^{(s)} | \Theta)$  has the form given in (2), the sum over  $\mathcal{K}$  has  $K^{\sum_s L_s}$  terms and  $p(\Theta) = p(\mathcal{T})p(\mathbf{B})p(\Sigma)$  is some prior.

**The complete log-likelihood** The EM algorithm aims at computing (the E-step) the expectation of the complete log-likelihood which is given by

$$\begin{aligned} Q(\Theta; \hat{\Theta}) &\equiv \mathbb{E} \left[ \log p(\mathcal{X}, \mathcal{K} | \Theta) | \mathcal{X}, \hat{\Theta} \right] \\ &= \underbrace{\sum_{s=1}^S \sum_{t=2}^{L_s} \sum_{l=1}^K \bar{w}_{t,l}^{(s)} \log \mathcal{N}(\mathbf{x}_t^{(s)} | \mathbf{x}_{t-1}^{(s)} + \mathbf{T}_l(\mathbf{x}_{t-1}^{(s)}), \sigma_l^2 \mathbf{I})}_{\bar{A}(\mathcal{X}, \mathcal{K})} \\ &\quad + \underbrace{\sum_{s=1}^S \sum_{t=2}^{L_s} \sum_{l=1}^K \sum_{g=1}^K \bar{w}_{t,g,l}^{(s)} \log B_{g,l}(\mathbf{x}_{t-1}^{(s)})}_{\bar{B}(\mathcal{X}, \mathcal{K})}, \end{aligned} \quad (5)$$

where  $\bar{w}_{t,l}^{(s)} = P[w_{t,l}^{(s)} = 1 | \mathbf{x}^{(s)}, \hat{\Theta}]$ , and  $\bar{w}_{t,g,l}^{(s)} = P[w_{t,g,l}^{(s)} | \mathbf{x}^{(s)}, \hat{\Theta}]$  which are obtained by a modified forward-backward procedure [3].

The M-step maximizes the Q-function in (5) with respect to the model parameters  $\Theta$ . The maximization with respect to the motion vector fields  $\mathcal{T}$  and noise variances  $\Sigma$  (the term  $\bar{A}(\mathcal{X}, \mathcal{K})$  in (5)) is straightforward and it follows the same strategy as in [1]. What is different is the way we perform the optimization of the switching probabilities, (the term  $\bar{B}(\mathcal{X}, \mathcal{K})$  in (5)) that we detail next.

**Natural gradient** Optimization of switching probabilities is usually dealt with as a constrained optimization problem since probabilities have to lie on a probability simplex. Here, we adopt a different approach based on the information geometric framework [8, 9]. Within this framework, switching probabilities are considered as points on a statistical manifold and their optimization is then performed as an unconstrained optimization problem on the manifold.

When dealing with categorical distributions, with probabilities  $(p_1, \dots, p_K)$ , one possible parameterization is to use only  $K-1$  probabilities  $\xi = [p_1 \dots p_{K-1}]^T$  as coordinates, the remaining probability  $p_K$  being dependent and automatically computed from the normalization constraint by  $p_K = 1 - \sum_{i=1}^{K-1} p_i$ . This parameterization provides a global coordinate system on the manifold. Then, a metric can be introduced by defining an inner product on the tangent space of each point  $\xi$ . A particularly interesting metric makes use of the Fisher information matrix  $G = [g_{ij}]$  defined by

$$g_{ij}(\xi) = E \left[ \frac{\partial \log p}{\partial \xi^i} \frac{\partial \log p}{\partial \xi^j} \right]. \quad (6)$$

This metric has the property that it is invariant with respect to coordinate transformations and is central to the definition of natural gradient.

Generically, the gradient of a function  $f$  is the vector  $\nabla f$  such that  $\langle \nabla f, v \rangle = df(v)$  holds for any vector  $v$ . In this equation,  $\langle \cdot, \cdot \rangle$  denotes an inner product and  $df$  the differential of the function  $f$  (a 1-form whose components are given by the partial derivatives of  $f$ ). In the matrix convention adopted henceforth, the differential is written as the row matrix of components  $df = \left[ \frac{\partial f}{\partial p_1} \dots \frac{\partial f}{\partial p_{k-1}} \right]$  and the gradient vector as a column matrix. Specializing the gradient vector computation for the statistical manifold of categorical probability distributions, and using the Fisher information metric (6) above, yields

$$\nabla f = G^{-1}(df)^T = \xi \circ (df)^T - \xi (df \cdot \xi), \quad (7)$$

where the operator  $\circ$  denotes the Hadamard product (element-wise product of vectors) and  $\cdot$  is the usual dot product. This gradient can be computed with linear time and memory complexity, avoiding the explicit construction of the Fisher information matrix and it can be shown [2] that  $\nabla f$  vanishes on the boundaries of the probability simplex. This latter fact alone turns probability optimization into an unconstrained problem, since it is not possible to move out of the simplex using an appropriately bounded, but positive, optimization step.

The maximization of a generic function  $f$  is then performed using the gradient method

$$\xi_{t+1} = \xi_t + \eta \nabla f, \quad (8)$$

where the scalar  $\eta$  is a positive step size. This equation is iterated until convergence is attained. To ensure that the update corresponds to a valid probability distribution, the step size has to satisfy the bounds

$$0 < \eta < \frac{1}{\xi^T \nabla f - \alpha}, \quad \alpha = \min\{0, (\nabla f)_1, \dots, (\nabla f)_{K-1}\}. \quad (9)$$

The application of the natural gradient algorithm to optimize the switching probabilities is performed directly on the function  $Q$ . To this end, each stochastic matrix  $\mathbf{b}^{(n)}$  is viewed as a collection of categorical distributions, one at each row, and each one is optimized independently of the others.

## 4 Experimental Results

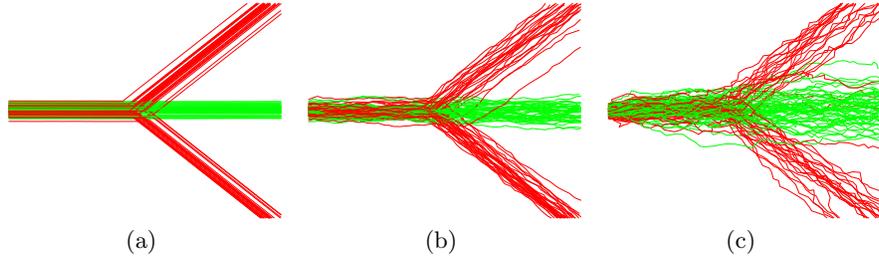
This section reports experimental results using the proposed approach to learn the generative model in (1) to perform trajectory classification. Furthermore, we also provide a comparative evaluation with the projection simplex.

In a classification context, we assume that we have a number  $A$  of different activity classes, i.e.  $a \in \{1, \dots, A\}$ , and that we have a subset of trajectories from each of these activity classes,  $\mathcal{X}^1, \dots, \mathcal{X}^A$ . We will denote the set of the fields and parameters corresponding to each activity class  $a$  as  $\theta^a = (\mathcal{T}^a, \mathbf{B}^a, \Sigma^a)$ , for  $a = 1, \dots, A$ . In some cases, one or more of these collections of parameters may be shared among the classes; for example, if the motion fields are common among the classes and only the switching matrices differ, we have  $\mathcal{T}^a = \mathcal{T}$  and  $\Sigma^a = \Sigma$ , for  $a = 1, \dots, A$ . We will illustrate the performance classification in both synthetic and real scenarios.

### 4.1 Synthetic data

In this example, two classes (activities) of trajectories are considered (see Fig. 1). One that contains one left-right motion model to describe, say, the *straight* activity. This activity has zero probability of switching, i.e. identity transition matrices everywhere in the grid. The second activity contains three motion models, one left-right horizontal motion as in the previous class, and two (up and down) diagonal motions to describe the *disperse* activity. In this case, the trajectory can turn up or down, thus we set in the middle of the image region. the diagonal of the transition matrix to 0.8 and the remaining entries to  $0.2/(K-1)$ , with  $K$ , the number of motion fields. All the trajectories start roughly at the image point  $[0, 0.5]^T$  of the unit square as illustrated in Fig. 1(a).

In this experiment we assumed the number of motion models previously known, thus  $K = 3$ . Nevertheless, this could be automatically be determined using a discriminative based approach as in [4]. Also, we perform a comparison



**Fig. 1.** Two synthetic trajectory classes. Straight (green) and disperse (red) activities containing different level of dynamic noise: (a)  $\sigma_{\text{tst}}^2 = 1\sigma_{\text{trn}}^2$ , (b)  $\sigma_{\text{tst}}^2 = 50\sigma_{\text{trn}}^2$  and (c)  $\sigma_{\text{tst}}^2 = 100\sigma_{\text{trn}}^2$

between the proposed method described in Section 3 and the gradient projection (GP) algorithm [5]. The two main components of the GP algorithm are: (i) the computation of the gradient of the objective function and the projection onto the constraint set (i.e. stochastic matrices). Concerning the latter (i.e. the projection), it consists in projecting each row of the transition matrix onto a probability simplex, for which a recent proposed approach has been proposed (see [6]).

To perform the comparison, 10 experiments were conducted. For each experiment, we generated 100 (see (1)) training and testing trajectories. For the training trajectories we set  $\sigma_{\text{trn}}^2 = 10^{-4}$ , for the testing set we used the following values for the dynamic noise:  $\sigma_{\text{tst}}^2 = \{\sigma_{\text{trn}}^2, 2\sigma_{\text{trn}}^2, 5\sigma_{\text{trn}}^2, 8\sigma_{\text{trn}}^2, 10\sigma_{\text{trn}}^2, 16\sigma_{\text{trn}}^2, 20\sigma_{\text{trn}}^2, 32\sigma_{\text{trn}}^2, 50\sigma_{\text{trn}}^2, 100\sigma_{\text{trn}}^2\}$  (each experiment contains 100 test trajectories with  $\sigma_{\text{tst}}^2$  set to a value in the above interval). With this strategy it is possible to verify the robustness of the proposed approach against trajectory mismatch. The performance is measured in terms of classification accuracy which is accomplished by simply using the forward E-step.

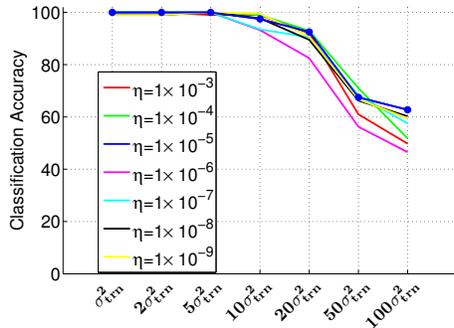
Fig. 1 shows the testing trajectories with (b)  $\sigma_{\text{tst}} = 32\sigma_{\text{trn}}$  and (c)  $\sigma_{\text{tst}} = 100\sigma_{\text{trn}}$ .

To perform a fair comparison, the initial conditions of the EM algorithm are the same for both of the methodologies. The parameter values are set as follows: we used 7 iterations for the EM,  $K = 3$  motion fields, and 10 iterations in the M-step to estimate the transition matrix  $\mathbf{B}$ . We performed 8 runs of the EM for both of the algorithms. The step size  $\eta$  is determined by cross validation in the interval  $\sigma_{\text{tst}}^2 = \{\sigma_{\text{trn}}^2, 2\sigma_{\text{trn}}^2, 5\sigma_{\text{trn}}^2, 10\sigma_{\text{trn}}^2, 20\sigma_{\text{trn}}^2, 50\sigma_{\text{trn}}^2, 100\sigma_{\text{trn}}^2\}$ . For both methodologies a value of  $\eta = 10^{-5}$  was found the most suitable in the interval values  $\{10^{-3}, \dots, 10^{-9}\}$ . Fig. 2(b) shows the performance of the natural gradient for each value of  $\sigma_{\text{tst}}^2$ , and for different values of  $\eta$  (see the dotted blue line in Fig. 2(b) for the best value of  $\eta$ ). Fig. 2(a) shows the initialization of the remaining parameters.

Fig. 3 shows the box plot graphs concerning the accuracy of the project simplex (a) vs. the proposed natural gradient (b), for the best 4 initializations of the EM. Here, we discarded the cases where the EM has deficient convergence

Parameter	Initial conditions
$\Sigma = \{\sigma_{(1)}^2, \dots, \sigma_{(N)}^2\}$	$\rightarrow 1 \times 10^{-3}$
$\mathcal{T} = \{\mathbf{t}_{(1)}, \dots, \mathbf{t}_{(N)}\}$	$\rightarrow$ random in $[-0.01 \ 0.01]$
$\mathcal{B} = \{\mathbf{b}_{(1)}, \dots, \mathbf{b}_{(N)}\}$	$\rightarrow \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$

(a)



(b)

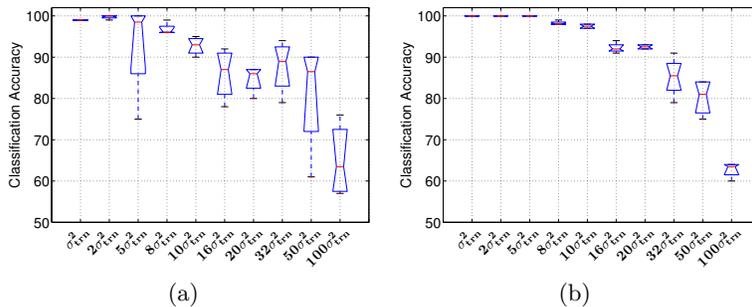
**Fig. 2.** Parameters initialization for the EM (a), selection of the  $\eta$  parameter for the natural gradient method (b).

due to the poor initializations. We see that the natural gradient provides better performance, exhibiting higher mean values and smaller error deviation. The PG algorithm has more instability through the interval values of  $\sigma_{\text{tst}}^2$ .

**Robustness to initialization** In the previous experiment we avoided the worst EM estimates, taking only the best initializations. Here, we provide an additional experiment to illustrate how robust are the frameworks to initialization. For that purpose, we provide the statistics performance for each trial that ranging from 4 up to 8 runs of the EM. Fig. 4 show the performance of the methodologies increasing the number of the EM runs. Although, both methods decrease the performance as the dynamic noise increases, the natural gradient provides better results in terms of the first (higher values) and second (smaller values) order statistics.

## 4.2 Real data

We now consider the application of the proposed algorithm in a real scenario: the campus of the Universitat Politcnica de Catalunya (UPC) Barcelona. In this scenario the images were obtained from a remote and fixed network camera located at the UPC campus. The camera was continuously streaming during 5 hours. Several classes of trajectories were observed and thus considered for



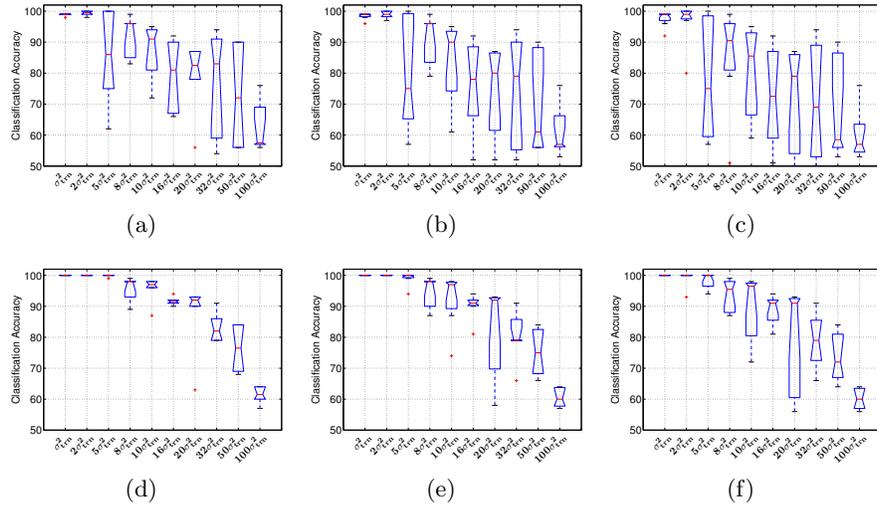
**Fig. 3.** Classification accuracy using (a) the projection simplex and (b) natural gradient.

classification. The trajectories were obtained by tracking the pedestrians using the LOTS algorithm [7]. Fig. 5 (a) depicts some of the trajectories after a homography transformation belonging to each activity (in a different color). The obtained trajectories are divided into the following activities (see Fig. 5 (a)): ( $a_1$ ) *walking and stepping down the stairs* (from up-left to bottom right direction); ( $a_2$ ) *walking along* (up motion); ( $a_3$ ) *crossing and stepping up the stairs*; ( $a_4$ ) *pass diagonally down*; ( $a_5$ ) *turning he Campus*.

In the synthetic example we assumed the number of fields to be known. In the real case, we have to automatically discover the most appropriate number of models. To accomplish this, we assume that all the activities *share* the same vector fields, i.e the class specific models are  $\Theta_K^{(a)} = (\mathcal{T}, \mathbf{B}^{(a)}, \Sigma)$ , for  $a \in \{1, \dots, A\}$ , where only the switching matrices differ among the classes, and  $K$  is the number of (shared) vector fields. To determine the model order, we consider, in addition to the training set, a *validation* or *selection* set  $\mathcal{D}$ . The model selection is then performed by assessing the classification accuracy with each model order in  $\mathcal{D}$ . In our experiments, we varied the the number of models in the interval  $K = \{1, \dots, 8\}$ . Using the set  $\mathcal{D}$  we concluded that  $K = 6$  provided the best classification accuracy. Fig. 5(b) shows the performance of the algorithm for different values of  $K$ . We see that the validation set generalizes well for the test set. Fig. 5(c) discriminates the accuracy in terms of trajectory classification among the considered classes for the best value of  $K$ .

## 5 Conclusions

This paper presented an innovative approach to the estimation of switching probabilities in the context of human activity recognition. Compared to other methods applied before to the same problem, the proposed algorithm has the advantage that the optimization is performed in the unconstrained realm, reducing the computational complexity of previous constrained optimization methods. The proposed approach was tested and validated both on synthetic data and real

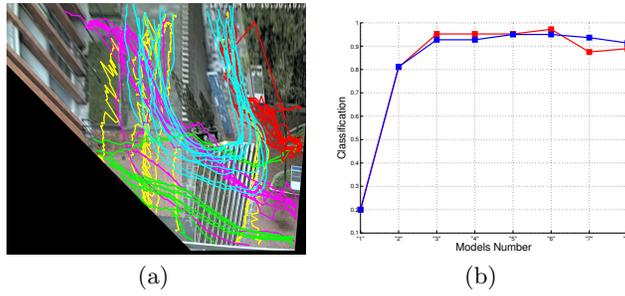


**Fig. 4.** Performance of the Projection simplex (top) and natural gradient (bottom), for 6 (left) 7 (middle) and 8 (right) runs of the EM.

data obtained from surveillance videos. It is shown empirically that the natural gradient method converges faster and attained better accuracy generically.

## References

1. J. C. Nascimento, M. A. T. Figueiredo and J. S. Marques “Activity recognition using mixture of vector fields”, *IEEE Trans. on Image processing*, 2012 (in press).
2. M. Barão and J. M. Lemos, “An efficient Kullback-Leibler optimization algorithm for probabilistic control design”, *Mediterranean Conf. on Control and Automation* pp. 198-203, 2008.
3. L. Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
4. J. C. Nascimento, J. S. Marques and M. A. T. Figueiredo, “Discriminative Model Selection Using a Modified Bayesian Criterion: Application to Trajectory Modeling”, *Int. Conf. Imag. Procesing*, 2011.
5. J. Nocedal and S. Wright, *Numerical Optimization*, Springer, New York, 2006.
6. J. Duchi, S. Shalev-Shwartz, and T. C. Y. Singer, “Efficient projections onto the  $\ell_1$ -ball for learning in high dimensions,” in *Proc. of Int. Conf. on Machine Learning*, Helsinki, Finland, 2008.
7. T. Boulton, R. Micheals, X. Gao, and M. Eckmann, “Into the woods: Visual surveillance of non-cooperative camouflaged targets in complex outdoor settings,” in *Proceedings of the IEEE*, pp. 1382–1402, 2001.
8. Shun-ichi Amari, “Natural gradient work efficiently in learning”, *Neural Computation*, Vol. 10, No. 2, pp. 251–276, 1998.
9. Shun-ichi Amari, Hiroshi Nagaoka, “Methods of Information Geometry”, Oxford University Press, 2000.



	$a_1(\%)$	$a_2(\%)$	$a_3(\%)$	$a_4(\%)$	$a_5(\%)$
$a_1$	99.2	0	0	0.8	0
$a_2$	0	91.9	8.1	0	0
$a_3$	0	0	83.3	0	16.7
$a_4$	0	0	0	100.0	0
$a_5$	0	0	0	0	100.0

(c)

**Fig. 5.** (a) Homography of the trajectories:  $a_1 \rightarrow$  walking and stepping down the stairs (pink - from up-left to bottom right direction);  $a_2 \rightarrow$  walking along (up motion in yellow);  $a_3 \rightarrow$  crossing and stepping up the stairs (green);  $a_4 \rightarrow$  pass diagonally down (red);  $a_5 \rightarrow$  turning the Campus (cyan); (b) performance accuracy in the validation (red) and test (blue) for  $K = \{1, \dots, 8\}$ , (c) Classification results of the five activities for the best values  $K = 6$ .