Sorting Atomic Activities for Discovering Spatio-temporal Patterns in Dynamic Scenes

Gloria Zen[1], Elisa Ricci[2], Stefano Messelodi[3], and Nicu Sebe[1]

[1] Department of Information Engineering and Computer Science, Università di Trento, Italy
{gloria.zen, nicu.sebe}@disi.unitn.it
[2] Department of Electronic and Information Engineering, Università di Perugia, Italy
elisa.ricci@diei.unipg.it
[3] Fondazione Bruno Kessler, Trento, Italy
messelod@fbk.eu

Abstract. We present a novel non-object centric approach for discovering activity patterns in dynamic scenes. We build on previous works on video scene understanding. We first compute simple visual cues and individuate elementary activities. Then we divide the video into clips, compute clip histograms and cluster them to discover spatio-temporal patterns. A recently proposed clustering algorithm, which uses as objective function the Earth Mover's Distance (EMD), is adopted. In this way the similarity among elementary activities is taken into account. This paper presents three crucial improvements with respect to previous works: (i) we consider a variant of EMD with a robust ground distance, (ii) clips are represented with circular histograms and an optimal bin order, reflecting the atomic activities'similarity, is automatically computed, (iii) the temporal dynamics of elementary activities is considered when clustering clips. Experimental results on publicly available datasets show that our method compares favorably with state-of-the-art approaches.

Keywords: Dynamic scene understanding, Earth Mover's Distance, Linear Programming, Dynamic Time Warping, Traveling Salesman Problem

1 Introduction

State-of-the-art video surveillance systems are currently based on an object centric perspective [1, 2], *i.e.* first interesting objects in the scene are detected and classified according to their nature (*e.g.* pedestrians, cars), then tracking algorithms are used to follow their paths. However these approaches are suboptimal when monitoring wide and complex scenes and it is essential to take into account the presence of several occlusions and the spatial and temporal correlations between many objects. Therefore, unsupervised non-object centric approaches for dynamic scene understanding have gained popularity in the last few years [3–6]. These methods use low level features (such as position, size and motion of pixels or blobs) to individuate elementary activities. Then, by analyzing the co-occurrences of events, high level activity patterns are discovered. Most of the recent approaches for complex scene analysis are based on Probabilistic Topic Models [3–5]. It has been shown that these methods are particularly effective to discover activity patterns and to model other interesting aspects such as the behaviors'correlation over time and space. However, since they rely on the traditional word-document paradigm for representing atomic activity distributions into clips, they discard any information about the similarity among elementary activities. To overcome this drawback recently Zen *et al.* [6] proposed a different approach. By optimizing a cross-bin distance function (*i.e.* EMD) rather than a bin-to-bin one, they showed that the problem of discovering high-level activity patterns in dynamic scenes can be modeled as a simple Linear Programming (LP) problem. To achieve scalability on large datasets they also propose a simplification of the optimization problem by establishing a words'order and considering only the similarity among adjacent words. However in [6] the order of atomic activities is chosen based on heuristics. In this paper, we propose a more rigorous strategy to sort atomic activities relying on the adaptation of the traveling salesman problem (TSP) to the task at hand. We also adopt a circular histogram representation for clips and optimize a variant of EMD with a robust ground distance. This allows us to improve the accuracy of clustering results at the expenses of a modest increase of the computational cost with respect to [6]. Here, as in many previous works [3–6], clips are represented by histograms. This has a beneficial effect in terms of filtering out noise. On the other hand any information about the temporal dynamics of atomic activities inside a clip is ignored. To compensate for this fact in this paper we propose to compute a dynamic time warping (DTW) similarity score between pairs of clips and to construct a nearest neighbor graph which is used to bias clips assignment toward appropriate clusters. The paper is organized as follows. In Section 2 the work in [6] is briefly summarized. Section 3 presents our approach for ordering atomic activities and the resulting LP problem. The proposed strategy for incorporating temporal information into the learning algorithm is also discussed. Experimental results are presented in Section 4. Finally, in Section 5 the conclusions are drawn.

2 Discovering Patterns with Earth Mover's Prototypes

The approach proposed in [6] is articulated in two phases. In the first phase simple motion features are extracted from the video and used to individuate elementary activities. In the second phase the video is divided into short clips and for each clip c an histogram $\boldsymbol{h}_c$ counting the occurrences of the elementary activities is computed. Then the clips are grouped according to their similarity and a small set of histogram prototypes representing typical activity patterns occurring in the scene are extracted. The Earth Mover's Prototypes learning algorithm in [6] amounts into solving the following optimization problem:

$$\min_{\boldsymbol{p}_i \geq 0,\ \sum_q p_i^q = 1} \sum_{i=1}^{N} \mathcal{D}_E(\boldsymbol{h}_i, \boldsymbol{p}_i) + \lambda \sum_{i \neq j} \eta_{ij} \max_{q=1\dots D} |p_i^q - p_j^q| \tag{1}$$

where $\{\boldsymbol{h}_1, \dots, \boldsymbol{h}_N\}$, $\boldsymbol{h}_i \in I\!R^D$, are the original clip histograms, $\{\boldsymbol{p}_1, \dots, \boldsymbol{p}_N\}$, $\boldsymbol{p}_i \in I\!R^D$, are the computed prototypes and $\mathcal{D}_E(\boldsymbol{h}, \boldsymbol{p})$ is the EMD [7]. The EMD among two normalized histograms (*i.e.* $\sum_q p^q = 1$ and $\sum_t h^t = 1$) is defined as:

$$\mathcal{D}_E(\boldsymbol{h}, \boldsymbol{p}) = \min_{f_{qt} \geq 0} \sum_{q,t=1}^{D} d_{qt} f_{qt} \quad \text{s.t.} \quad \sum_{q=1}^{D} f_{qt} = h^t, \ \sum_{t=1}^{D} f_{qt} = p^q \tag{2}$$

Solving (1) the prototypes $\boldsymbol{p}_i$ are computed in order to maximize their similarity with respect to the original histograms $\boldsymbol{h}_i$. At the same time the number of different prototypes is imposed to be small by minimizing their reciprocal differences. The relative importance of the two requirements is controlled by the positive coefficient λ. The coefficients $\eta_{ij} \in \{0, 1\}$ are fixed and are used to select the pairs of histograms which must be merged. In practice, by substituting the definition (2) into (1), the following LP is obtained:

$$\begin{aligned} \min_{p_i^q, f_{qt}^i, \zeta_{ij} \geq 0} \ & \sum_{i=1}^{N} \sum_{q,t=1}^{D} d_{qt} f_{qt}^i + \lambda \sum_{i \neq j} \eta_{ij} \zeta_{ij} \\ \text{s.t.} \quad & -\zeta_{ij} \leq p_i^q - p_j^q \leq \zeta_{ij}, \ \forall q, \forall i, j = 1 \ldots N, \quad i \neq j \\ & \sum_{q=1}^{D} f_{qt}^i = h_i^t, \ \forall t, \ \forall i = 1 \ldots N \qquad \sum_{t=1}^{D} f_{qt}^i = p_i^q, \ \forall q, \forall i = 1 \ldots N \end{aligned} \tag{3}$$

where the slack variables ζ_{ij} are introduced.

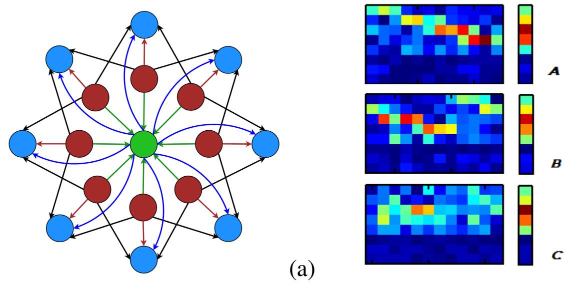
A nice characteristic of (3) is that the ground distances d_{qt} can encode information about the similarity between atomic activities. However this flexibility comes at the expenses of a considerable computational cost. This cost is especially high due to the large number of flow variables f_{qt}^i, which is quadratic in the size of histograms D. Therefore, in order to speed up calculations, a modification of (3) which adopts EMD with L_1 distance over bins as ground distance (*i.e.* $d_{qt} = |q - t|$) is proposed in [6]. In this case, referred as EMD-L_1, the optimization problem simplifies and the number of flow variables reduces from $O(ND^2)$ to $O(ND)$ [8]. The idea is that similar atomic activities should correspond to neighboring bins in activity histograms. To this aim the atomic activities are sorted according to the associated location and motion information. However in [6] simple heuristics are used in this phase. In the following section we present a more rigorous approach to sort atomic activities.

3 Circular Earth Mover's Prototypes

In this Section we present the main phases of our approach, starting from low level features extraction to high-level activity patterns discovery by computing Circular Earth Mover's Prototypes.

3.1 Computing Atomic Activities

Given an input video, first low level features are extracted. We apply a background subtraction algorithm [9] to extract pixels of foreground. For these pixels we also compute the optical flow vectors using the Lucas-Kanade algorithm. By thresholding the



(a) (b)

Fig. 1. (Best viewed in color.) **(a)** An example of the flow network associated to (4). Two circular histograms with $D = 8$ bins are compared. The green node is the transhipment vertex. Ingoing edge (green) cost is the threshold (2 in this case) and outgoing edge (blue) cost is 0. Red edges have cost 0. The black edges are 1-cost edges. **(b)** The importance of considering the temporal dynamics of atomic activities when comparing clips. For three clips (A,B,C) the sequences of frame histograms (left) and the associated average clip histograms (right) are shown. While all the average histograms are similar, only clip A and clip B should be assigned to the same cluster. In fact the temporal dynamics of atomic activities is very different in clip C.

magnitude of the optical flow vectors, foreground pixels are classified into static and moving pixels. Moving foreground pixels are also classified based on the direction of the optical flow vector. We consider 8 possible directions. Then we divide the scene into $p \times q$ patches. For each patch a descriptor vector $\boldsymbol{v} = [x\ y\ f_g\ \bar{d}_{of}\ \bar{m}_{of}]$ is computed where (x, y) denotes the coordinates of the patch center in the image plane, f_g represents the percentage of foreground pixels and $\bar{d}_{of}$ and $\bar{m}_{of}$ are respectively the mode of the orientations distribution and the average magnitude of optical flow vectors. For patches of static pixels we set $\bar{d}_{of} = \bar{m}_{of} = 0$. Valid patches are only those with $f_g \geq T_{fg}$. Then the K-medoids algorithm is applied to the set of valid patch descriptors $\boldsymbol{w} = [x\ y\ \bar{d}_{of}\ \bar{m}_{of}]$ and a dictionary of atomic activities is constructed.

3.2 Ordering Atomic Activities

The K-medoids algorithm provides a set of D centroids $\boldsymbol{c}^d = [x^d\ y^d\ \bar{d}^d_{of}\ \bar{m}^d_{of}]$ representing typical elementary activities. However elementary activities are not independent and it is desirable to take into account their correlation when learning activity prototypes. To this aim in [6] the authors proposed to order atomic activities in a way that neighboring activities correspond to similar ones. While in [6] the order is determined based on simple heuristics, in this paper we propose to adopt a TSP strategy to sort atomic activities. The TSP [10] is a well known combinatorial optimization problem. The goal is to find the shortest closed tour connecting a given set of cities, subject to the constraint that each city must be visited only once. In this paper we model the task of computing the optimal order of atomic activities as the problem of computing the optimal city tour.

Formally, the TSP can be stated as follows. A distance matrix $\mathbf{D}$ with elements d_{qt}, $q, t = 1, \ldots, D$ and $d_{qq} = 0$ is given. The value d_{qt} represents the distance between the q-th and the t-th city. A city tour can be represented by a cyclic permutation π of

$\{1, 2, \ldots, D\}$ where $\pi(q)$ represents the city that follows the city q on the tour. The TSP is the problem of finding a permutation π that minimizes the length of the tour $\ell = \sum_{q=1}^{D} d_{q\pi(q)}$. In this paper we consider the symmetric TSP (*i.e.* $d_{qt} = d_{tq}$) and a metric as intercity distance, *i.e.* we set $d_{qt} = \mathcal{D}(\boldsymbol{c}^q, \boldsymbol{c}^t) = \sqrt{(x^q - x^t)^2 + (y^q - y^t)^2} + \gamma\sqrt{(\bar{d}^q_{of} - \bar{d}^t_{of})^2 + (\bar{m}^q_{of} - \bar{m}^t_{of})^2}$. The parameter γ is used to balance the importance of the position and the motion information.

The TSP can also be formulated as a graph theoretic problem. Given a complete graph $G = (V, E)$ the cities correspond to the node set $V = \{1, 2, \ldots, D\}$ and each edge $e_{qt} \in E$ has an associated weight d_{qt}. The TSP amounts to find a Hamiltonian cycle, *i.e.* a cycle which visits each node in the graph exactly once, with the least weight in the graph. For this task, the tour length of $(D-1)!$ permutation vectors have to be compared. This results in a problem which is known to be NP-complete. However there are several heuristic algorithms for solving the symmetric TSP. In this paper we use a combination of the Christofides heuristic [11] for tour construction and simulated annealing for tour improvement.

3.3 Discovering Circular Earth Mover's Prototypes

To discover activity prototypes the video is divided into short clips and for each clip c a circular histogram $\boldsymbol{h}_c$ is created with bin orders obtained by solving the TSP. Finally, the clips are grouped according to their similarity and a set of circular histogram prototypes $\boldsymbol{p}_i$ is computed. They represent the salient activities occurring in the scene.

To this aim we solve (3). However, in this paper we use as ground distance a thresholded modulo L_1 distance [12] *i.e.* we set $d_{qt} = \min(\min(|q-t|, D - |q-t|), 2)$. The adoption of this ground distance with respect to the L_1 distance proposed in [6] allows us to deal in a principled way with circular histograms at the expenses of a modest increase of the computational cost. In fact, as shown [13], the adoption of a thresholded distance implies the introduction of a transhipment vertex, with slight increase of the number of flow variables. Moreover, it has been shown that saturated distances are beneficial in terms of accuracy results in several applications. With thresholded ground distance, the EMD (2) assumes the form:

$$\begin{aligned} &\min_{f_{q,q+1}, f_{q,q-1}, f_{q,D+1} \geq 0} \quad \sum_{q=1}^{D} f_{q,q+1} + \sum_{q=1}^{D} f_{q,q-1} + 2\sum_{q=1}^{D} f_{q,D+1} \\ &\text{s.t.} \quad f_{q,q+1} - f_{q+1,q} + f_{q,q-1} - f_{q-1,q} + f_{q,D+1} = h^q - p^q \quad \forall q \end{aligned} \tag{4}$$

where the flow variables $f_{q,D+1}$ correspond to the links connecting sources to the transhipment vertex. Figure 1.a depicts the associated flow network. In practice with respect to (2), in (4) only flows between neighbor bins and flows between sources and the transhipment vertex are considered. The number of flow variables is still $O(D)$. Note also that since histograms are circular $q+1 = 1$ if $q = D$ and $q-1 = D$ if $q = 1$.

The proposed prototype learning algorithm is obtained substituting the EMD definition (4) into (1), *i.e.* solving:

$$\min \sum_{i=1}^{N}\sum_{q=1}^{D} f^i_{q,q+1} + \sum_{i=1}^{N}\sum_{q=1}^{D} f^i_{q,q-1} + 2\sum_{i=1}^{N}\sum_{q=1}^{D} f^i_{q,D+1} + \lambda \sum_{i\neq j} \eta_{ij}\zeta_{ij} \tag{5}$$
$$\text{s.t.}\ \ -\zeta_{ij} \leq p^q_i - p^q_j \leq \zeta_{ij},\ \forall q,\ \forall i,j,\ i\neq j$$
$$f^i_{q,q+1} - f^i_{q+1,q} + f^i_{q,q-1} - f^i_{q-1,q} + f^i_{q,D+1} = h^q_i - p^q_i,\ \forall q, i$$
$$p^q_i, f^i_{q,q+1}, f^i_{q,q-1}, f^i_{q,D+1}, \zeta_{ij} \geq 0$$

The resulting optimization problem is a LP with $n_{var} = 4ND + \frac{1}{2}N(N-1)$ variables if we impose each prototype to be close to each other, *i.e.* $\eta_{ij} = 1\ \forall i \neq j$. Therefore the number of variables is slightly larger than in EMD-L_1 [6] where $n_{var} = 2N(D-1) + ND + \frac{1}{2}N(N-1)$. However (5) allows us to deal with circular histograms and to consider an optimal order of atomic activities. This provides more accurate clustering results as shown in the experimental section.

3.4 Embedding Temporal Information into Clustering

A nice characteristic of (5) is that, thanks to the introduction of the binary coefficients $\eta_{ij} \in \{0,1\}$, it is possible to select the pairs of histograms which must be merged. Generally a comparison among all possible pairs $\{\boldsymbol{p}_i, \boldsymbol{p}_j\}$, $i \neq j$, is required, imposing all prototypes to be close to each other. However by choosing only few $\eta_{ij} = 1$ it is possible to embed into the clustering algorithm some a-priori knowledge about the subset of histograms which must be fused. For example in [6], for each histogram $\boldsymbol{h}_i$ the set of P nearest neighbors is identified and $\eta_{ij} = 1$ if $\boldsymbol{h}_j$ is a neighbor of $\boldsymbol{h}_i$. This has the effect of producing a biased clustering assignment which is imposed to reflect the structure of the nearest neighbor graph. Moreover it greatly reduces the computational cost of solving (2) since the number of slack variables (*i.e.* constraints) is limited.

While in [6] a simple Euclidean distance is adopted, in this paper we present a better strategy to create a nearest neighbor graph. We propose to compute the distance among clips by taking into account the temporal dynamics of atomic activities inside the clip. More specifically for each clip c we consider not only the average histogram $\boldsymbol{h}_c$ but also the sequence of histograms $H_c = \{\mathbf{h}^1_c, \dots \mathbf{h}^M_c\}$ where $\mathbf{h}^i_c$ is the histogram of elementary activities computed on the i-th frame. Then, to construct the nearest neighbor graph, we propose to adopt a function which measures the distance between two histogram sequences considering the match of their alignment. This allows us to account for small shifts inside a clip and to consider two clips as similar only if the activity patterns inside them have a similar temporal structure. This concept is exemplified in Fig.1.b. In particular in this paper we consider the dynamic time warping and longest common subsequence (LCSS) distances [14].

DTW. Given two clips H_a and H_b and the set $\mathcal{A}$ of all possible alignments ρ between them, the DTW distance is defined as:

$$D_{DTW}(H_a, H_b) = \min_{\rho \in \mathcal{A}(H_a,H_b)} \sum_{i=1}^{|\rho|} \kappa(\mathbf{h}^{\rho(i)}_a, \mathbf{h}^{\rho(i)}_b)$$

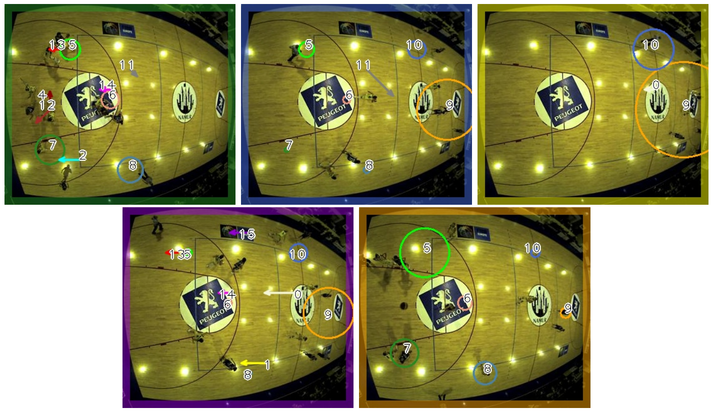


Fig. 2. APIDIS dataset: typical activities automatically discovered solving (5).

where $\kappa(\cdot)$ is the L_1 distance between histograms. Dynamic programming is used to compute the DTW distance, *i.e.* the optimal alignment between the two sequences of histograms.

LCSS. LCSS is also an alignment tool but is more robust to noise and outliers than DTW because not all points need to be matched. Instead of a one-to-one mapping between points, a point with no good match can be ignored to prevent unfair biasing. The LCSS distance is defined as:

$$D_{LCSS}(H_a, H_b) = 1 - \frac{LCSS(H_a, H_b)}{M}$$

As for DTW, dynamic programming can be used to compute LCSS, *i.e.* :

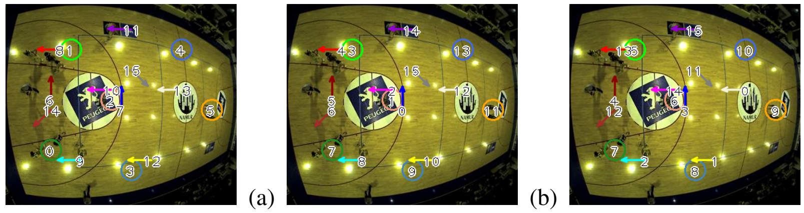
$$LCSS(H_a, H_b) = \begin{cases} 0, & m = 0 \mid n = 0 \\ 1 + LCSS(H_a^{m-1}, H_b^{n-1}), & \kappa(\mathbf{h}_a^m, \mathbf{h}_b^n) \leq \epsilon,\ |n - m| < \delta; \\ \max(LCSS(H_a^{m-1}, H_b^n), LCSS(H_a^m, H_b^{n-1})), & \text{otherwise} \end{cases}$$

4 Experimental Results

We tested the proposed approach on two publicly available datasets. Our method is fully implemented in C++ using the libraries OpenCV and GLPK 4.2.1 (GNU Linear Programming Kit) as the backend linear programming solver.

The first dataset is taken from **APIDIS**[4] and consists in a video sequence where players involved in a basketball match are depicted. A sequence of 3000 frames is chosen. The patch size is set to 16×16 pixels and each clip contains 60 frames, corresponding to a time interval of about 3 sec. The number of atomic activities is fixed to

[4] http://www.apidis.org/Dataset/



(a) (b) (c)

Fig. 3. APIDIS dataset: different orders of atomic activities. (a) Heuristics. (b) TSP (only position). (c) TSP (position and motion).

Table 1. APIDIS dataset: clustering accuracy $(\%)$ of the proposed approach with different orders of atomic activities

Random	Heuristics	TSP (only position)	TSP (position and motion)
78	80	80	**88**

16. Solving (5) we automatically identify the five main activities occurring in the scene. Figure 2 shows a representative frame for each of them: (i) the blue team is attacking while the yellow team is on defence (green), (ii) the players are moving away from the yellow team's court side (blue), (iii) the blue team is on the defence (yellow), (iv) the players are moving back towards the yellow team's side (violet) and (v) the players of the blue team are shooting free throws (orange). Similar activities were also discovered in [6]. Table 1 shows some results of a quantitative evaluation of our method. Here the cluster assignments obtained solving (5) are compared with the ground truth build as in [6], based on the event annotation taken from the APIDIS website. The clustering performance corresponding to different ways of ordering atomic activities are compared. It is straightforward to observe that the TSP method outperforms other strategies. Moreover both motion and position information are crucial for obtaining accurate results (γ is set to 0.5). Figure 3 depicts an example of different orders of atomic activities. In this case the order based on the heuristics corresponds to sort activities considering first static ones ordered according to their position along the x axis, then those with motion different from zero.

The second dataset[5] [3] depicts a complex traffic scenes with cars moving in proximity of a **Roundabout**. It corresponds to a video of about 1 hour duration (93500 frames, 25 Hz framerate). In our experiment we consider only the first 30 minutes in order to compare quantitatively our results with those provided in [6, 15]. In this case the patch size is set to 12×12 pixels while histograms of activities have 16 bins. Fig. 4 depicts an example of the typical activities discovered for this dataset, corresponding mainly to a horizontal and a vertical flow of vehicles. Table 2 shows the results of a quantitative evaluation. The proposed algorithm outperforms state-of-the-art approaches *e.g.* EMD-L_1 and L_1 methods [6] and PLSA and hierarchical PLSA [15]. The associated color bars depicting the results of temporal segmentation are shown in Fig. 4 (bottom). As observed for the APIDIS dataset, choosing a suitable order of atomic

[5] http://www.eecs.qmul.ac.uk/∼jianli/Roundabout.html

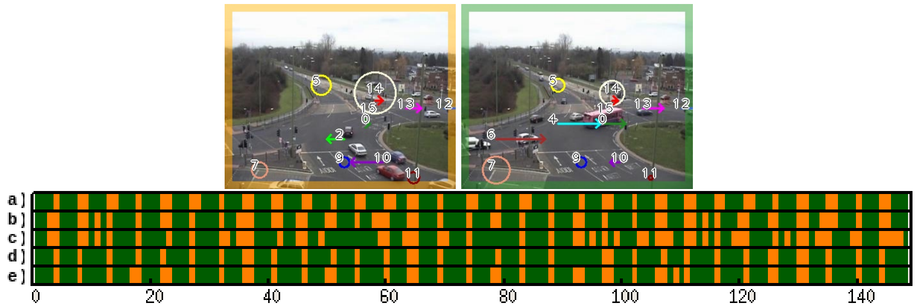


Fig. 4. Roundabout dataset. (top) Typical activities automatically discovered solving (5). (bottom) Temporal segmentation: (a) Ground truth, (b) Hierarchical PLSA [15], (c) Standard PLSA [15], (d) Our appraoch (5) - LCSS (TSP), (e) EMD-L_1 [6]

Table 2. Roundabout dataset: Clustering performance

method	accuracy (%)
L_1 [6]	86.4
EMD-L_1 (random) [6]	72.3
EMD-L_1 (heuristics) [6]	86.4
(5) - DTW (random)	81.63
(5) - LCSS (random)	83
(5) - DTW (TSP)	**87.75**
(5) - LCSS (TSP)	**87.75**
Standard PLSA [15]	84.46
Hierarchical PLSA [15]	72.30

activities is crucial: using a random order the performance decrease significantly. Moreover a TSP strategy is also desirable with respect to an approach based on heuristics. Finally the adoption of DTW and LCSS distances for setting the coefficient η_{ij} further improves the clustering results. In fact, a better nearest neighbor graph drives the clustering algorithm towards more accurate solutions. Some videos showing the results of our experiments can be found at *http://tev.fbk.eu/people/ricci/iciap2011.html*.

5 Conclusions

We presented a novel method for discovering spatio-temporal patterns in dynamic scenes. Differently from most of the previous works on non-object centric dynamic scene analysis, our approach provides a principled way to deal with similarity of elementary activities while learning high-level activity prototypes. It relies on an automatical way to compute the optimal order of atomic activities and to an adaptation of the clustering algorithm in [6] to take into account the temporal dynamics of atomic activities inside the clips. Many interesting aspects still deserve study. For example more sophisticated mechanisms to filter out the noise of low level features must be exploited. On a theoretical side, we are currently investigating an approach for learning the ground distances.

References

1. Wang X., Tieu, K., Grimson, W.E.L.: Learning Semantic Scene Models by Trajectory Analysis. European Conference on Computer Vision (ECCV), 2006.
2. Zelniker, E.E., Gong, S.G., Xiang, T.: Global Abnormal Detection Using a Network of CCTV Cameras. Workshop on Visual Surveillance, 2008.
3. Hospedales, T., Gong, S., Xiang, T.: A Markov Clustering Topic Model for Mining Behaviour in Video. IEEE International Conference on Computer Vision (ICCV), 2009.
4. Varadarajan, J., Emonet, R., Odobez, J.-M.: Probabilistic Latent Sequential Motifs: Discovering temporal activity patterns in video scenes. British Machine Vision Conference (BMVC), 2010.
5. Kuettel, D., Breitenstein, M. D., Van Gool, L., Ferrari, V.: What's going on? Discovering Spatio-Temporal Dependencies in Dynamic Scenes. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010.
6. Zen, G., Ricci, E.: Earth Mover's Prototypes: a Convex Learning Approach for Discovering Activity Patterns in Dynamic Scenes. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
7. Rubner, Y., Tomasi, C., Guibas, L.J.: The Earth Movers Distance as a Metric for Image Retrieval. International Journal of Computer Vision (IJCV), 40, 2, 99–121, (2000).
8. Ling, H., Okada, K.: An efficient earth movers distance algorithm for robust histogram comparison. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 29, 5, 840–853 (2006).
9. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1999.
10. Lawler, E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B.: The Traveling Salesman Problem. Wiley, New York (1985).
11. Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA (1976).
12. Pele, O., Werman, M.: A linear time histogram metric for improved sift matching. European Conference on Computer Vision (ECCV), 2008.
13. Pele, O., Werman, M.: Fast and robust earth movers distances. IEEE International Conference on Computer Vision (ICCV), 2009.
14. Rabiner, L., Juang, B.: Fundamentals of Speech Recognition. Englewood Cliffs, New Jersey, Prentice-Hall (1993).
15. Li, J., Gong, S., Xiang, T.: Global Behaviour Inference using Probabilistic Latent Semantic Analysis. British Machine Vision Conference (BMVC), 2008.