Tracking System with Re-identification Using a Graph Kernels Approach

Amal Mahboubi, Luc Brun, Donatello Conte, Pasquale Foggia, Mario Vento

Tracking System with Re-identification using a Graph Kernels Approach

Amal Mahboubi[1], Luc Brun[1],
Donatello Conte[2], Pasquale Foggia[2], and Mario Vento[2]

[1] GREYC UMR CNRS 6072, Equipe Image ENSICAEN
6, boulevard Maréchal Juin F-14050 Caen, FRANCE,
amal.mahboubi@unicaen.fr , luc.brun@ensicaen.fr

[2] Dipartimento di Ingegneria dell'Informazione, Ingegneria Elettrica e Matematica Applicata
Università di Salerno, Via Ponte Don Melillo, 1 I-84084 Fisciano (SA), ITALY
dconte@unisa.it , pfoggia@unisa.it , mvento@unisa.it

Abstract. This paper addresses people re-identification problem for visual surveillance applications. Our approach is based on a rich description of each occurrence of a person thanks to a graph encoding of its salient points. People appearance in a video is encoded by bags of graphs whose similarities are encoded by a graph kernel. Such similarities combined with a tracking system allow us to distinguish a new person from a re-entering one into a video. The efficiency of our method is demonstrated through experiments.

Keywords: Visual surveillance, Graph Kernel, Re-identification

1 Introduction

Re-identification is a recent field of study in pattern recognition. The purpose of re-identification is to identify object/person coming back onto the field view of a camera. Such a framework may be extended to the tracking of object/persons on a network of cameras.

Methods dealing with the re-identification problem can be divided into two categories. A first group is based on building a unique signature for object. Features used to describe signatures are different: regions, Haar-like features, interest points [1], [2]. The second group of methods [3], [4] does not use a single signature for the object, but the latter is represented by a set of signatures. Thus, the comparison between objects takes place between two sets of signatures rather than between two individual signatures.

The basic idea of our work starts from the consideration that there are few works that exploit relationships between the visual features of an object. Furthermore, our work combines both approaches by describing a person both with a global descriptor over several frames and a set of representative frames. More

precisely, the principle of our approach is to represent each occurrence of a person at time t by a graph representation called a t-prototype (Section 2). A kernel between t-prototypes (Section 3) is proposed in order to encode the similarity between two persons based on their appearance on a single frame.

The design of the proposed kernel in Section 3 is based on a previous kernel [6] devoted to image indexation. However for people re-identification, tracking problems have to be addressed in order to cover the re-identification investigations. Within our framework a person is not characterized by a single image but by a sequence of images encoding its appearance along several frames. This new proposal (as shown by the dotted box in Figure 1) is described in section 4 and 5. The global appearance of a person over a video, is described by a bag of t-prototypes (Section 4) and global features of the bag computed on representative t-prototypes. The temporal window over which is build a bag of t-prototypes is called the history tracking window (HTW). Kernels between bags of t-prototypes are proposed in Section 4.1 in order to measure the similarity of two persons on several frames. Such kernels are used within our tracking system (Section 5) in order to determine if an entering person is a new person or a re-entering one. The efficiency of the proposed approach is evaluated through experimental results in section 6.

2 T-prototype construction

The first step of our method consists to separate subjects from the background. To that end, we use binary object masks [5] defined by a foreground detection with shadow removals. Each moving person within a frame is thus associated to a mask that we characterize using SIFT key point detectors. Such key points provide a fine local characterization of the image inside the mask which is robust against usual image transformations such as scales and rotations. Each key-point is represented by its x and y coordinates, scale, orientation and 128 numbers (the descriptors) per color channel. In order to contextualize the information encoded by SIFT points we encode them by a mutual k nearest neighbor graph $G = (V, E, w)$ where V corresponds to the set of SIFT points, E to the set of edges and w is a weight function defined over V and defined as the scale of appearance of the corresponding vertex. The set of edges E is defined from the key point coordinates x and y: one edge (v, v') belongs to E if v belongs to the k nearest neighbors of v' while v' belongs to the k nearest neighbors of v. The degree of each vertex is thus bounded by k. For a given vertex u, we take into account the local arrangement of its incident vertices by explicitly encoding the sequence of its neighbors encountered when turning counterclockwise around it. This neighborhood $\mathcal{N}(u) = (u_1, ..., u_n)$ is thus defined as an ordered set of vertices. The first vertex of this sequence u_1 is arbitrary chosen as the upper right vertex. The set $\{\mathcal{N}(u)\}_{u \in V}$ is called the bag of oriented neighborhoods (BON). The node u is called the central node.

3 Kernel between t-prototypes

Our kernel between t-prototypes (eq. 5) is based on a previous contribution [6] within the image indexation framework. This kernel is based on the description of each graph by a finite bag of patterns. Such an approach consists to: i) define the bag of patterns from each graph, ii) define a minor kernel between patterns, iii) convolve minor kernels into a major one in order to encode the similarity between bags. SIFT points being local detectors, we consider that the more relevant information of a t-prototype corresponds to the local oriented neighborhood of its vertices. We thus define the bag of patterns of a t-prototype as its BON (section 2). The minor kernel between oriented neighborhoods is defined as follows:

$$K_{seq}(u,v) = \begin{cases} 0 & if\ |\mathcal{N}(u)| \neq |\mathcal{N}(v)| \\ \prod_{i=1}^{|\mathcal{N}(u)|} K_g(u_i, v_i)\ otherwise \end{cases} \quad (1)$$

where $K_g(u,v)$ is a RBF kernel between features of input vertices defined by a tuning parameter σ and the Euclidean distance $d(.,.)$ between feature values: $K_g(x,y) = e^{-\frac{d(\mu(x),\mu(y))}{\sigma}}$.

Eq. 1 corresponds to the same basic idea that the heuristic used to compute the graph edit distance between two nodes [7] where the similarity between two nodes is enforced by a comparison of their neighborhoods.

Note that $K_{seq}(.,.)$ corresponds to a tensor product kernel and is hence definite positive. However, due to acquisition noise or small changes between two images, some SIFT points may be added or removed within the neighborhood of some vertices. Such an alteration of the neighborhood's cardinal may drastically change the similarity between key points. Indeed, according to equation (1), two points with a different neighborhood's cardinal have a similarity equal to 0. Equation (1) induces thus an important sensibility to noise. In order to overcome this drawback, we introduce a rewriting rule on oriented neighborhoods. Given a vertex v, the rewriting of its oriented neighborhood denoted $\kappa(v)$ is defined as: $\kappa(v) = (v_1, ..., \widehat{v_i}, ..., v_{l_v})$ where $\widehat{v_i} = argmin_{j \in \{1,...,l_v\}} w(v_j)$ is the neighbor of v with lowest weight.

This rewriting is iterated leading to a sequence of oriented neighborhoods $(\kappa^i(v))_{i \in \{0,...,D_v\}}$, where D_v denotes the maximal number of rewritings. The cost of each rewriting is measured by the cumulative weight function CW defined by:

$$\begin{cases} CW(v) & = 0 \\ CW(\kappa^i(v)) & = w(v_i) + CW(\kappa^{i-1}(v)) \end{cases} \quad (2)$$

where v_i is the vertex removed between $\kappa^{i-1}(v)$ and $\kappa^i(v)$.

Kernel between oriented neighborhoods: Our kernel between two oriented neighborhoods is defined as a convolution kernel between the sequence of rewritings of each neighborhood, each rewriting being weighted by its cumulative cost:

$$K_{rewriting}(u,v) = \sum_{i=1}^{D_v} \sum_{j=1}^{D_u} K_W(\kappa^i(u), \kappa^j(v)) * K_{seq}(\kappa^i(u), \kappa^j(v)) \quad (3)$$

where kernel K_W penalizes costly rewritings corresponding to the removal of important key-points. Such a kernel is defined as follows:

$$K_W(\kappa^i(u), \kappa^j(v)) = e^{-\frac{CW(\kappa^i(u))+CW(\kappa^j(v))}{\sigma'}} \text{ where } \sigma' \text{ is a tuning variable.} \quad (4)$$

The number of rewritings (D_v) for each vertex v corresponds to a compromise between an over simplification of its oriented neighborhood (large D_v) and the corruption of equation 3 by non relevant vertices which may appear in only one of two similar oriented neighborhoods. This number has been empirically set to half the cardinal of v's neighborhood [6].

Graph Kernel: Taking into account central nodes, our final kernel between two vertices u and v is defined as follows: $K(u, v) = K_g(u, v)K_{rewriting}(u, v)$

Our final kernel between two graphs is defined as a convolution kernel between both BONs:

$$K_{graph}(G_1, G_2) = \sum_{u \in V_1} \sum_{v \in V_2} \varphi(u)\varphi(v)K(u, v) \quad (5)$$

The weighting function φ encodes the relevance of each vertex and is defined as an increasing function of the weight: $\varphi(u) = e^{-\frac{1}{\sigma'(1+w(u))}}$

4 People description

The identification of a person by a single t-prototype is subject to errors due to slight changes of the pose or some errors on the location of SIFT points. Assuming that the appearance of a person remains stable on a set of successive frames, we describe a person at instant t by the set of its t-prototypes computed on its HTW window. The description of a person, by a set of t-prototypes provides an implicit definition of the mean appearance of this person over HTW. Let $\mathcal{H}$ denotes the Hilbert space defined by K_{graph} (equation 5). In order to get an explicit representation of this mean appearance, we first use K_{graph} to project the mapping of all t-prototypes onto the unit-sphere of $\mathcal{H}$. This operation is performed by normalizing our kernel [8]. Following [8], we then apply a one class ν-SVM on each set of t-prototypes describing a person. From a geometrical point of view, this operation is equivalent to model the set of projected t-prototypes by a spherical cap defined by a weight vector w and an offset ρ both provided by the ν-SVM algorithm. These two parameters define the hyper plane whose intersection with the unit sphere defines the spherical cap. T-prototypes whose projection on the unit sphere lies outside the spherical cap are considered as outliers. Each person is thus encoded by a triplet (w, ρ, S) where S corresponds to the set of t-prototypes and (w, ρ) are defined from a one class ν-SVM. The parameter w indicates the center of the spherical cap and may be intuitively understood as the vector encoding the mean appearance of a person over its HTW window. The parameter ρ influence the radius of the spherical cap and may be understand as the extend of the set of representatives t-prototypes in S.

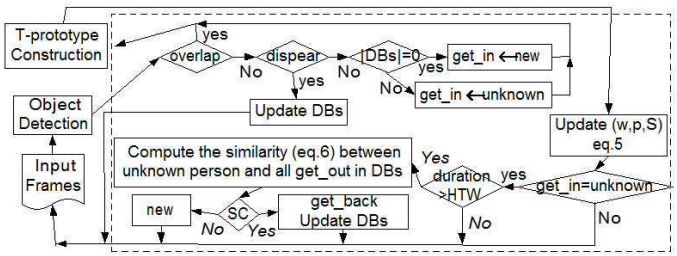

Fig. 1. Algorithm steps

4.1 People's Kernel

Let $P_A = (w_A, \rho_A, S_A)$ and $P_B = (w_B, \rho_B, S_B)$ denote two triplets encoding two persons A and B. The distance between A and B is defined from the angle between vectors w_A and w_B defined by [8] as follows:
$d_{sphere}(w_A, w_B) = arccos\left(\frac{w_A^T K_{A,B} w_B}{\|w_A\|\|w_B\|}\right)$ where $\|w_A\|$ and $\|w_B\|$ denote the norms of w_A and w_B in $\mathcal{H}$ and $K_{A,B}$ is a $|S_A| \times |S_B|$ matrix defined by $K_{A,B} = (K_{norm}(t,t'))_{(t,t') \in S_A \times S_B}$, where K_{norm} denotes our normalized kernel. Based on d_{sphere}, the kernel between A and B is defined as the following product of RBF kernels:

$$K_{change}(P_A, P_B) = e^{\frac{-d^2_{sphere}(w_A, w_B)}{2\sigma^2_{moy}}} e^{\frac{-(\rho_A - \rho_B)^2}{2\sigma^2_{origin}}} \quad (6)$$

Where σ_{moy} and σ_{origin} are tuning variables.

5 Tracking system

Our tracking algorithm uses four labels 'new', 'get_out', 'unknown' and 'get_back' with the following meaning: *new* refers to an object classified as new, *get-out* represents an object leaving the scene, *unknown* describes a query object (an object recently appeared, not yet classified) and *get-back* refers to an object classified as an old one.

Unlike our previous work [5], where we used a training data set to model each object and the re-identification was triggered by an edit graph distance. in this paper, we are using online learning and the re-identification is performed using the similarity (eq.6) between each unknown person and all the get_out persons. The general architecture of our system is shown in Figure 1. All masks detected in the first frame of a video are considered as new persons. Then a mask detected in frame $t+1$ is considered as matched if there is a sufficient overlap between its bounding box and a single mask's bounding box defined in frame t. In this case, the mask is affected to the same person than in frame t and its graph of SIFT points is added to the sliding HTW window containing the last graphs of this person. If one mask defined at frame t does not have

any successor in frame $t+1$, the associated person is marked as get_out and its triplet $P = (w, \rho, S)$ (Section 4) computed over the last $|HTW|$ frames is stored in an output object data base model noted DB_S. In the case of a person corresponding to an unmatched mask in frame $t+1$, the unmatched person is initially labeled as 'get_in'. When a 'get_in' person is detected, if there is no 'get_out' persons we classify this 'get_in' person immediately as new. This 'get_in' person is then tracked along the video using the previously described protocol. On the other hand, if there is at least one 'get_out' person we should delay the identification of this 'get_in' person which is thus labeled as 'unknown'. This 'unknown' person is then tracked on $|HTW|$ frames in order to obtain its description by a triplet (w, ρ, S). Using this description we compute the value of kernel K_{change} (equation 6) between this unknown person and all get_out persons contained in our database. Similarities between the unknown person and get_out ones are sorted in decreasing order so that the first get_out person of this list corresponds to the best candidate for a re identification. Our criterion to map an unknown person to a get_out one, and thus to classify it as get_back is based both on a threshold on the maximum similarity values max_{ker} and a threshold on the standard deviations σ_{ker} of the list of similarities. This criterion called, SC is defined as $max_{ker} > th_1$ and $\sigma_{Ker} > th_2$, where th_1 and th_2 are experimentally fixed thresholds. Note that, SC is reduced to a fixed threshold on max_{ker} when the set of get_out persons is reduced to two elements. An unknown person whose SC criterion is false is labeled as a new person. Both new and get_back persons are tracked between frames until they get_out from the video and reach the get_out state.

Classically, any tracking algorithm has to deal with many difficulties such as occlusions. The type of occlusions examined in this paper is limited to the case where bounding boxes overlap. An occlusion is detected when the spatial overlap between two bounding boxes is greater than an experimentally fixed threshold while each individual box remains detected. If for a given object an occlusion is detected, the description of this object is compromised. Thus a compromised object is only tracked and its triplet (w, ρ, S) is neither updated nor stored in DB_S. At identification time, the model of the unknown person is matched against each get-out person from DB_S.

6 Experiments

The proposed algorithm has been tested on v01, v05, v04 and v06 video sequences of the PETS'09 S2L1 [9] dataset. Each sequence contains multiple persons. To compare our framework with previous work, we use the well-known metrics Sequence Frame Detection Accuracy ($SFDA$), Multiple Object Detection Accuracy ($MODA$) and Multiple Object Tracking Accuracy ($MOTA$) described in [11]. Note that such a measure does not allow to take into account the fact that the identification of a person may be delayed. Since our method identifies a person only after HTW frames, we decided not to take into account persons with

an unknown status in the $MODA$ and $MOTA$ measures until these persons are identified as get_back or new (Section 5).

In our first experiment we have evaluated how different values of the length of HTW may affect the re-identification accuracy. The obtained results show that, v01 and v05 perform at peak efficiency for HTW=35. V04 and v06 attain their optimum at HTW=20.

To validate our method of re-identification we used the Cumulative Matching Characteristic (CMC) curves. The CMC curve represents the percentage of times the correct identity match is found in the first n matches. Figure 2 shows the CMC curves for the four views. We can see that the performance of v01 is much better than that of v05, v06 and v04. We attribute this to the high detection accuracy in v01. Figure 2 shows that if we focus on the first 5 matches, we find that for v04 and v06 a score of 54% and 65% respectively is obtained, while for v01 and v05 it attains 100%.

In order to compare our results to the state of the art's methods we used the exhaustive comparison of 13 methods defined in [9]; where a quantitative performance evaluation of the submitted results by contributing authors of the two PETS workshops in 2009 on PETS'09 S2.L1 dataset was performed. Using the metrics MODA, MOTA, MODP, MOTP SODA and SFDA described in [11], the submitted results of [10] outperform all other methods. We hence only compare our results to this best method. The left column of Table 1 shows the best results [10] obtained by methods described in [9] on each video. As shown by the two left-most data columns of Table 1, our method obtains lower result than that of [10] for v04 and v06. This may be explained by the fact that v04 and v06 have persistent group cases. Indeed the case where two or more existing objects at time t become too spatially close at time $t+1$ and then merge together to become a one detected object at time $t+1$ is not considered here as an occlusion, but rather as a group. Since such case is not addressed by this paper, v04 and v06 results need to be interpreted with caution. Due to the frequent group cases in v04 and v06 we missed a lot of persons in the scene. However, our method obtains better result than that of [10] for v01 and v05. These results set forth the relevance of the proposed re-identification algorithm since we have only occlusion cases.

7 Conclusion

In this paper, we presented a new people re-identification approach based on graph kernels. Our graph kernel between SIFT points includes rewriting rules on oriented neighborhood in order to reduce the lack of stability of the key point detection methods. Furthermore, each person in the video is defined by a set of graphs with a similarity measure between sets which removes outliers. Our tracking system is based on a simple matching criterion to follow one person along a video. Person's description and kernel between these descriptions is used to remove ambiguities when one person reappears in the video. Such a system may be easily extended to follow one person over a network of camera. People

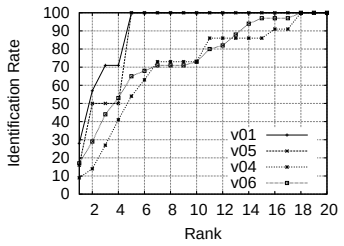

Fig. 2. CMC curves

Table 1. Evaluation results

View	MODA of[10]	MODA	MOTA	SFDA
v01	0.67	0.91	0.91	0.90
v05	0.72	0.75	0.75	0.80
v04	0.61	0.2799	0.2790	0.47
v06	0.75	0.506	0.505	0.64

are prone to occlusions by others nearby. However, a re-identification algorithm for an individual person is not suitable for solving the groups cases. A further study with more focus on groups is therefore suggested.

References

1. Hamdoun, O., Moutarde, F., Stanciulescu, B.,Steux, B.: Person re-identification in multi-camera system by signature based on interest point descriptors collected on short video sequences ICDSC 2008 1–6 (2008)
2. Y. Ijiri, S. Lao, T. X. Han and H. Murase. Human Re-identification through Distance Metric Learning based on Jensen-Shannon Kernel. VISAPP'2012, 603–612 (2012)
3. Truong Cong, D-N., Khoudour,L., Achard, C., Meurie, C., Lezoray,.O: People re-identification by spectral classification of silhouettes International Journal of Signal Processing 90, 2362–2374 (2010)
4. S. Zhao, F. Precioso and M. Cord. Spatio-Temporal Tube data representation and Kernel design for SVM-based video object retrieval system. *Multimedia Tools Appl.*, (55):105-125 (2011)
5. Brun, L., Conte, D., Foggia, P., Vento, M.: People Re-identification by Graph Kernel Methods GBR'11, 285-294 (2011)
6. Mahboubi, A., Brun, L., Dupé, F-X.: Object Classification Based On Graph Kernels HPCS-PAR 385-389 (2010)
7. S. Fankhauser, K. Riesen and H. Bunke. Speeding Up Graph Edit Distance Computation through Fast Bipartite Matching GBR'11, 102-111 (2011)
8. Desobry, F., Davy, M., Doncarli, C.: An Online Kernel Change Detection Algorithm IEEE Transaction on Signal Processing 53, 2961–2974 (2005)
9. Ellis, A., Shahrokni, A., Ferryman, J.: PETS 2009 and Winter PETS 2009 Results, a Combined Evaluation 12th IEEE Int. Work. on Performance Evaluation of Tracking and Surveillance 1–8 (2009)
10. J. Berclaz, A. Shahrokni, F. Fleuret, J.M. Freyman and P. Fua, Evaluation of probabilistic occupancy map people detection for surveillance systems, 11th IEEE Int. Work. on Performance Evaluation of Tracking and Surveillance, 55–62 (2009)
11. R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova and Jing Zhang. Framework for performance evaluation of face, text, and vehicule detection and tracking in video: Data, metrics, and protocol, *Pattern Analysis and Machine Intelligence*,IEEE Transaction on, 31(2):319-336, Feb. 2009.