Transductive People Tracking in Unconstrained Surveillance

Dalia Coppi *Student Member IEEE* Simone Calderara *Member IEEE*
and Rita Cucchiara *Member IEEE*

***Abstract*—Long term tracking of people in unconstrained scenarios is still an open problem due to the absence of constant elements in the problem setting. The camera, when active, may move and both the background and the target appearance may change abruptly leading to the inadequacy of most standard tracking techniques. We propose to exploit a learning approach that considers the tracking task as a semi supervised learning (SSL) problem. Given few target samples the aim is to search the target occurrences in the video stream re-interpreting the problem as label propagation on a similarity graph. We propose a solution based on graph transduction that works iteratively frame by frame. Additionally, in order to avoid drifting, we introduce an update strategy based on an evolutionary clustering technique that chooses the visual templates that better describe target appearance evolving the model during the processing of the video. Since we model people appearance by means of covariance matrices on color and gradient information our framework is directly related to structure learning on Riemannian manifolds. Tests on publicly available datasets and comparisons with state-of-the-art techniques allow to conclude that our solution exhibit interesting performances in terms of tracking precision and recall in most of the considered scenarios.**

***Index Terms*—Transductive Learning, People Tracking, Video-surveillance.**

I. Introduction

People visual tracking is challenging and it concerns most of the surveillance application: it is not only important by its own, but represents the preliminary step of high level analysis such as behaviors or actions recognition. Although the complexities of visual tracking have been identified in literature, most of the modern solutions fail to robustly achieve satisfying performances in all the possible conditions, [49]. Nevertheless, competitive results emerged under specific premises, such as a high target resolution, a peculiar motion model or possibly an a-priori known target shape [10], [13], [46], [43]. In surveillance applications, people tracking difficulties are mainly due to the typical low resolution of target images, the articulated human shape, the variable motion patterns and the heterogeneous sensors employed for the video acquisition that may be either fixed or moving cameras and finally wearable sensors. With the definition *tracking in unconstrained scenarios* we consider this kind of situation where no a-priori constraints are given on the type of camera, the occlusions, the lighting condition, the clutter circumstances or the target motion patterns. To this aim, we address the problem in a general environment when time/spatial coherency is not completely guaranteed (*e.g.* the target is not visible in every frame), when the motion of the target is unpredictable and its appearance may change significantly (*e.g.* the target takes off a coat). Removing the target motion constraints, we restate the single target tracking as a semi-supervised label propagation problem. Initial target detections are considered as the labeled data points while the unlabeled data are people patches extracted from the subsequent video frames. We propose to exploit graph based transductive learning where the nodes of the graph represent the people patches and edges represent the similarity between them and only few nodes of the graph are labelled. The tracking is consequently formulated as the problem of label propagation from labelled to unlabelled nodes of the graph.

The tracking process becomes consequently an iterative approach where people is detected frame-by-frame, possibly using a people detector [15], and inputed to the transduction optimization algorithm. To improve the discriminative power of the transductive learner, we exploit two different models of labeled instances, one representing the target and the other representing non-target elements. Drift effects are mitigated by an update strategy based on evolutionary spectral clustering to retain the best target images and smoothly model its appearance variability. This mechanism explicitly avoids drifting errors and is a novel aspect of our proposal. Eventually, among the possible plethora of descriptors we choose a covariance matrix descriptor because of its capability of integrating information concerning colours, textures and shape (edges) in a single compact and highly discriminative formulation. Moreover, covariance matrices represent points on a lower dimensional manifold where the graph Laplacian operator of the tranductive learner aims at learning its structure thus connecting in a joint framework both the features and the label optimization strategy. The latter aspect is also novel, to our knowledge, because most of the methods consider the features and the label assignment problem independently. Nevertheless, the proposed tracking scheme is not limited to covariance matrices but can be applied to any features representation lying on a Riemannian space.

II. Related works

A. Tracking Background

In the past, many tracking methods have relied on stable background subtraction from one or several static cameras, [9], [31], [62], [23], [29]. However, recent progress in automatic object detection have driven the attention to a different and more flexible approach to people tracking, [1], [61], [34].
The combination of object detection and a temporal assignment strategy results in algorithms that tend to be more

Authors are with the Department of Engineering "Enzo Ferrari"
University of Modena and Reggio Emilia
Via Vignolese 905/b Modena, Italy
email:{dalia.coppi,simone.calderara,rita.cucchiara}@unimore.it

suitable for on-line applications. This methodology is referred in literature as *tracking by detection*, [1], [5], [24], [34], [36], [45], [61], [32]. When computational capacity is available, often a Particle Filter (PF) is used to scan the whole target pdf described by either a parametric model or a non parametric density estimator, [1], [32], [48], [55], [44], [8], [33].
The shortcoming of PF is its effectiveness mainly on short sequences. Additionally, when the initial bounding box is sloppy, generative methods have empirically proved to perform worse; searching for the most probable match is likely to work well in tracking when occlusions and confusions occur, and in low contrast images. On the contrary, when the object is detectable (e.g. in the case of people tracking), probabilistic matching has no advantage over direct matching, [49]. In contrast to the generative PF approach, discriminative classification is used for tracking [57], [2], [26]. Discriminative tracking opens the door to robustness in general tracking as very large feature sets can be employed which have been proved successful in general object classification. The importance of the adopted classifier is motivated by the few initial training examples that are fed to the tracking system. Therefore, due to the lack of training data, simpler models such as nearest neighbor classifiers or discriminant analysis may appear more effective overall than Multiple Instance Learning, [43]. A fundamental problem is that wrongly labeled elements and an improper target model update scheme may confuse the classifier and lead to drifting.

Oppositely to tracking by detection approaches, that locally search for the best target assignment over two or a few consecutive frames, data association trackers exploit a global optimization strategy to link detection responses, or tracklets, to long trajectories. In these approaches objects are tracked until they are visible and at a different level of processing tracklets are associated, the objective is thus to globally optimize a set of detected trajectories. This approach is typically interpreted as an a-posteriori optimization method where, given the data extracted from a video sequence, a specific optimization method concurs linking data along the time axis using tracklets [42], [36] and possibly dealing explicitly with objects split and merge behavior [45], [27]. Classical approaches process the video sequence off-line and aims at optimizing a global association cost by either discriminative [7] or generative [51], [39] multiple hypotheses pruning.

B. Tracking as a learning problem

In most of the revised method, an object or motion model is either learnt or defined to solve the tracking problem. If this is helpful to improve the tracker accuracy might sometimes limit its capability in terms of generalization. Specifically, learning appearance models of the targets results in an excellent tracking method when the temporal coherence among the target visual features is maintained but often fails when this constraint is not satisfied and the target appearance changes. Model-free tracking has thus been introduced and aims at learning specific target classifiers without relying on target models, [50], and the tracking problem is interpreted as a classification problem where the knowledge about the target itself can be explicitly specified or learnt during the tracking process. Hence, the choice of the classifier is crucial to achieve good performance. Commonly used classifiers include adaptive classifiers [50] and on-line boosting [31]. The drawback is that model free trackers are subject to drifting problems, Matthews *et al.* [40] referred to this problem as *template update problem*. In order to mitigate this effect additional knowledge may be exploited, *e.g.* geometric verification, [18], combination of generative and discriminative models, co-learning using different types of features, [52], or constrained updates, [30]. Nevertheless, recent surveys on model-free tracking state the tracking problem is a semi-supervised learning problem, [49] but only few methods explicitly acknowledged this classification strategy, [14], [41], [60]. Our proposal instead relies on modeling the tracking problem as a transductive learning label propagation problem, with a limited number of constraints on both the target motion and appearance. The approach allows the exploitation of both target and non-target images to improve the tracking data association process. Additionally, among semi-supervised classification schemes, transduction evaluates both the relation among labeled and unlabeled elements; this has proven to be empirically more effective in practice.
Transductive learning (TL) has been introduced by Vapnik in 90's [54] and has evolved over previous decades as an effective technique for solving several Computer Vision problems. TL has been applied as a method for interactive image segmentation [38], in conjunction with transfer learning for actions, [20], and faces recognition, [35], for image retrieval and user relevance feedback, [25], [6]. Surprisingly, the TL paradigm has been weakly explored to solve the people tracking problem. Wu *et al.* [59] introduced the TL as a solution to the severe variation of the models in color tracking. They fitted the TL problem into an EM frameworks to estimate the pixel labels in hand and face color tracking. Zha *et al.* [60] proposed an on-line single target tracking using graph transduction applied to faces and cars but without considering the template update problem. Coppi *et al.* [14] proposed an on-line single target tracking method based on a graph based formulation of the TL problem. They prevent drifting by clustering the target images in the last k-frames and using these images to update the set of the labelled instances. Nevertheless their proposal neglects any temporal smoothness in the model update process, thus breaking the guarantee that target images lie on the same smooth Riemannian manifold.

Our approach, instead, tracks object using a spectral graph transducer formulation in conjuction with a smooth model updating scheme based on evolutionary clustering. Our proposal exhibits consistent performances and a strong theoretical relation with incremental manifold learning. This solution aims to exploit the spectral properties of a similarity graph, built over a complete set of data, to learn the manifold structure from which the data have been sampled, [37]. This intuition holds since graph Laplacian eigenvalues converge to the discretization of the Laplace Beltrami operator applied over the manifold and then are able to describe the functionals that regulate its structure.

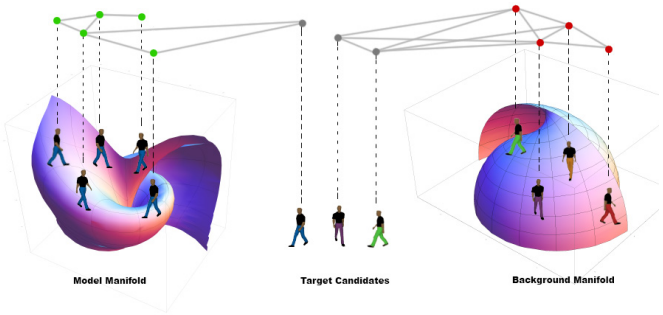


Fig. 1. Target samples in time help discovering the manifold that represents the target model. Locally, as the number of target examples increases, the planes tangent to the manifold are discovered and globally the manifold structure. Target appearance changes (e.g. front/back images) unveil different zone of the target manifold. Red points are adversal targets (non-target) that typically tend to be far from the target and help in the definition of the target manifold through dissimilarities. In the picture the positive model manifold is depicted on the right side while the negative elements are on the leftside. In the depicted graphs only the strongest edges (in terms of geodesic similarities) between input elements and the models are highlighted.

III. People tracking using semi supervised transductive learning

People tracking is partially different from general object tracking. Most of the scenarios where people tracking is applied are from surveillance setups where resolution is often low and the definition and details of the target are mostly poor. At the same time, in this scenario, object detection is a valuable help in distinguish among the targets and the background. Moreover, at low resolution people differences among people are helpful to distinguish the tracked target even in complex and cluttered situations. Following this idea, we consider the tracking process as a manifold learning problem where samples from the target manifold are discovered frame by frame, and at the same time samples from outside the manifold, and possibly lying on different disjoints manifolds, are available, Fig. 1. Every time new samples are discovered, new knowledge can be inserted into the data association process, introducing new constraints among input elements. More precisely, when the tracker is initialized we acquire at the same time information about the target and non-targets as well. Formally, we suppose to have at disposal a set of labeled instances $X^L = \{(x_i, y_i)\}$ where x_i are the input elements described by their features and $y_i = \pm 1$ their corresponding labels. In our setting, the input of our algorithm are the patches extracted from the first frame and the labels indicate whether those patches refer to the target object ($y_i = 1$) or to the background ($y_i = -1$). Besides, for each subsequent frame f of the considered sequence, suppose to dispose of a way to extract a set of unlabeled patches $X_f^U = \{(x_i)\}$.
We propose an iterative approach where we try to predict the target bounding box *frame-by-frame*, so that, at each step f, the complete dataset X_f comprises both the model X^L and the candidates samples X_f^U. By construction, the problem can now be regarded as expanding to each frame the information encoded into the model, which can be equivalently interpreted as the problem of propagating the labels from labeled to unlabeled examples in order to infer discriminative knowledge over the complete dataset.

People detection and representation

So far we assumed that the input of our framework are the patches of the people in the scene, therefore the first step should consist in people extraction from each frame of the video sequence. Although the tool we propose is basically independent from the specific people detection method we decide to extract people in the scene using a state-of-the-art people detector based on Histogram of Oriented Gradient, HOG, [15]. Dollar *et al.* [16] stated how detectors based on sliding windows appears more promising for low to medium resolution settings, which are the typical settings in video surveillance application, with respect to segmentation or keypoint methods that tend to fail under these conditions. Once the people bounding boxes are extracted from each frame, a predefined descriptor is computed to represent each snapshot and a specific metric is needed to match different occurrences of the same person, providing a reliable comparison between multiple samples.
Several descriptors have been proposed in literature (*e.g.* [17]) and various methods that combine shape, color and location exist. We adopted a covariance matrix descriptor [47], motivated by the possibility of combining color, shape and position cues without paying the additional computational complexity introduced by more complex methods based on part-based models. The same metric has been previously adopted in [42], [3] because of its robustness in matching the region in different views and poses. Covariance matrices exhibit scale and rotation invariance properties and are independent to changes in the average pixels intensity such as identical shifting of color values, i.e. changes in color due to shadows.

The covariance matrix is a square symmetric matrix $d \times d$, where d is the number of selected features independently from the size of the image window, carrying the advantage of being a low dimensional data representation. Given the covariance matrix C its diagonal entries represent the variance of each feature and the non-diagonal entries represent the correlations. Considering I as a three-dimensional color image and F as the $W \times H \times d$ dimensional feature image extracted from I,

$$F(x, y) = \Phi(I, x, y) \tag{1}$$

where the function Φ can be any mapping such as intensity, color, gradients, filter responses, etc. Let $\{z_i\}_{i=1\ldots N}$ be the d-dimensional feature points inside F, with $N = W \times H$. The image I is represented with the $d \times d$ covariance matrix of the feature points:

$$C_R = \frac{1}{N-1} \sum_{i=1}^{n} (z_i - \mu)(z_i - \mu)^T \tag{2}$$

where μ is the vector of the means of the corresponding features for the points within the region R.
In our case z_i is the feature vector composed for each pixel

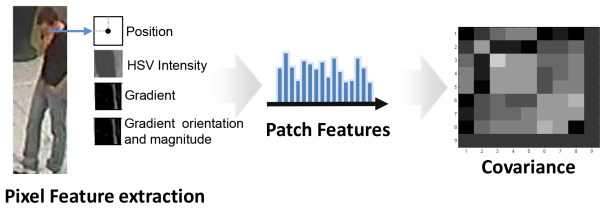


Fig. 2. The Figure shows an example of the features employed in the covariance matrix computation. Pixels position inside the people patch, H, S and V components in the HSV color space, Gradient x and y component and gradient magnitude and angles are considered. The final descriptor is a 9×9 covariance matrix.

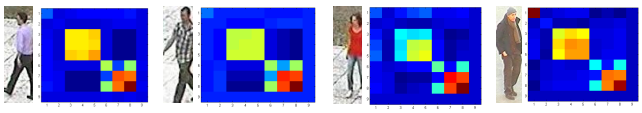


Fig. 3. People patches examples with their associated covariance matrix. Even with small patches the matrices appear visually different.

by its spatial, color and edge information. We use x and y pixel location in the image grid, HSV color values, G_x and G_y first order derivatives of the intensities calculated through Sobel operator w.r.t. x and y, and the magnitude $\|(x,y)\| = \sqrt{G_x^2 + G_y^2}$ and the angle $\theta(x,y) = \arctan\left(\frac{G_y}{G_x}\right)$ of the first order derivatives. Therefore each pixel of the image is mapped to a nine-dimensional feature vector

$$z_i = [\ x\ \ y\ \ H\ \ S\ \ V\ \ G_x\ \ G_y\ \ \|(x,y)\|\ \ \theta(x,y)\ ]^T \quad (3)$$

Based on this features vector the covariance of a region is a 9×9 matrix, as represented in Fig. 2.
It should be noted that we use HSV color space instead of the basic RGB color space because we experimented an higher invariance to scale and light changes.

Beyond the adopted feature vector, an adequate distance between covariance matrices must be defined to assess the appearance similarity between candidates regions and the target. However, the covariance matrices do not lie on the Euclidean space and arithmetic subtractions or simple operations between matrices are not meaningful. A robust distance metric between the covariance matrices is proposed in [21] as the sum of the squared logarithms of the generalized eigenvalues:

$$\rho(C_i, C_j) = \sqrt{\sum_{i=1}^{d} \ln^2 \lambda_k(C_i, C_j)} \quad (4)$$

where $\lambda_k(C_i, C_j)_{k=1\ldots d}$ are the generalized eigenvalues of C_i and C_j computed as:

$$\lambda_k C_1 x_k - C_j x_k = 0 \quad k = 0 \ldots d \quad (5)$$

where x_k are the generalized eigenvectors. The distance measure ρ satisfies the metric axioms, positivity, symmetry, triangle inequality, for positive definite symmetric matrices.

Spectral Graph Transducer for tracking

Let $G = (A, X_f)$ be the undirected graph at a specific frame f, where the nodes represent the n patches in our dataset and the edges are encoded by $A = \{a_{ij}\}$, a similarity matrix between the elements in X_f, derived from Eq. (4). The objective of the transductive learning algorithm is to find a cut of the graph that separates positive and negative elements, X^+ and X^- respectively, in such a way that the labelled examples X^L are correctly separated and, at the same time, patches with high similarity end up in the same cluster. The first one of these two postulates underlines the need for a cut consistent with the knowledge encoded in the model, while the latter is needed in order to exploit the meaningful structure of the data when considered in a similarity space.

It is known that there are many ways to partition a graph while still satisfying the above requirements, so a regularization scheme is needed in order to obtain a unique solution. In particular, if we know that the number of patches belonging to the object and to the background are equally distributed, then we should be able to find a partition which generates two balanced sets. This specific cut is often referred to as ratiocut. Considering D as the diagonal degree matrix $D_{ii} = \sum_j A_{ji}$, $L = D^{-1}(D - A)$ the normalized laplacian and ignoring the constraints on the labels, the ratiocut optimization problem can be stated as

$$\min_{\vec{\mathbf{z}}} \frac{\vec{\mathbf{z}}^T L \vec{\mathbf{z}}}{\vec{\mathbf{z}}^T \vec{\mathbf{z}}} \qquad \text{with } z_i \in \{\gamma_+, \gamma_-\} \quad (6)$$

where $\gamma_+ = \sqrt{\frac{|\{i:z_i<0\}|}{|\{i:z_i>0\}|}}$ and $\gamma_- = \sqrt{\frac{|\{i:z_i>0\}|}{|\{i:z_i<0\}|}}$. It is straightforward to verify that $\vec{\mathbf{z}}^T \vec{\mathbf{z}} = n$ and $\vec{\mathbf{z}}^T \mathbf{1} = 0$ for every feasible point. While this problem is still NP-hard, the minimum of its real relaxation

$$\begin{aligned} &\min_{\vec{\mathbf{z}}} \vec{\mathbf{z}}^T L \vec{\mathbf{z}} \\ &\text{s.t.}\ \vec{\mathbf{z}}^T \vec{\mathbf{z}} = n \text{ and } \vec{\mathbf{z}}^T \mathbf{1} = 0. \end{aligned} \quad (7)$$

is equal to the second eigenvalue of L and the corresponding eigenvector is the solution [53]. In order to take into account the labeling constraints a quadratic penalty term can be introduced in the objective function, so as to obtain

$$\begin{aligned} &\min_{\vec{\mathbf{z}}} \vec{\mathbf{z}}^T L \vec{\mathbf{z}} + C(\vec{\mathbf{z}} - \vec{\gamma})^T \mathrm{I} (\vec{\mathbf{z}} - \vec{\gamma}) \\ &\text{s.t.}\ \vec{\mathbf{z}}^T \vec{\mathbf{z}} = n \text{ and } \vec{\mathbf{z}}^T \mathbf{1} = 0 \end{aligned} \quad (8)$$

For each labelled patch, the corresponding element of $\vec{\gamma}$ is equal to $\tilde{\gamma}_+ (\tilde{\gamma}_-)$ for positive and negative examples, and it is zero for test examples. Of course $\tilde{\gamma}_+$ and $\tilde{\gamma}_+$ are estimates and can be computed based on the number of observed positive and negative patches in the training set. Note that in order to balance reliability of the training set over the feature similarity a trade-off constant C has been introduced.

The optimization problem of Eq. (8) can be recast as a quadratic eigenvalue problem (QEP) and solved analytically for positive semi-definite matrices as shown by Tisseur and Meerbergen [53]. Specifically, given the eigendecomposition $L = U \Sigma U^T$ of the normalized laplacian, a new parameter $\vec{\mathbf{w}}$ can be introduced and by substituting $\vec{\mathbf{z}} = U \vec{\mathbf{w}}$ and recalling that the smallest eigenvalue associated eigenvector is always

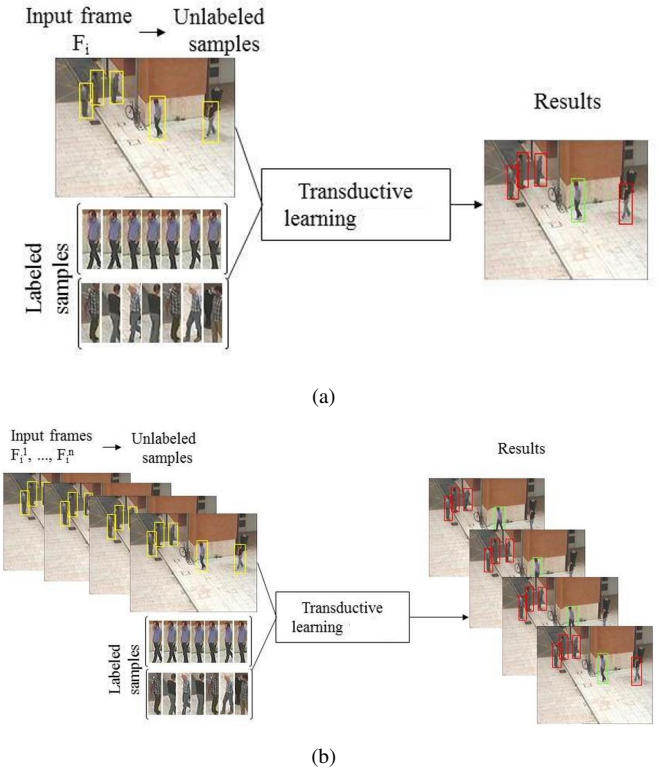


Fig. 4. Single (a) and Multiple (b) frame transduction. In the single frame transduction only one frame F_i is employed, in the multiple frame transduction a set $\{F_i^1, \ldots, F_i^m\}$ of frames is employed. Yellow rectangles on the frames on the left side represent the unlabeled input detection. On the right side red rectangles represent the samples classified as "no target" and green rectangles represent the samples classified as "target" by the TL. Labeled samples under the input frames are respectively the positive (target model), first row and negative model, second row.

$\mathbf{1}$, the second constraint of OP (7) reduce to $w_1 = 0$. Redefine V and Λ as the matrices containing, respectively, all the eigenvalues in Σ except the first one and all the eigenvectors associated to V, Eq. (8) is rewritten as

$$\begin{aligned} &\min_{\overrightarrow{\mathbf{w}}} \overrightarrow{\mathbf{w}}^T \Lambda \overrightarrow{\mathbf{w}} + C(V\overrightarrow{\mathbf{w}} - \overrightarrow{\gamma})^T \mathrm{I}(V\overrightarrow{\mathbf{w}} - \overrightarrow{\gamma}) \\ &\text{s.t.} \overrightarrow{\mathbf{w}}^T \overrightarrow{\mathbf{w}} = n. \end{aligned} \tag{9}$$

Finally, by introducing the terms $H = (\Lambda + CV^T V)$ and $\overrightarrow{\mathbf{b}} = CV^T \mathrm{I}\overrightarrow{\gamma}$ the objective function can be one more time reformulated, disregarding constant terms with respect to the optimization variable, as $\overrightarrow{\mathbf{w}}^T G \overrightarrow{\mathbf{w}} - 2b\overrightarrow{\mathbf{b}}^T \overrightarrow{\mathbf{w}}$. Following the *Courant-Fischer-Weyl min-max principle* the minimization of Eq. (9) is then solved for $\overrightarrow{\mathbf{w}} = (G - \lambda^* \mathrm{I})^{-1} \overrightarrow{\mathbf{b}}$, where λ^* is the smallest eigenvalue of

$$\begin{bmatrix} G & -\mathrm{I} \\ \frac{1}{n}\overrightarrow{\mathbf{b}}\overrightarrow{\mathbf{b}}^T & G \end{bmatrix}. \tag{10}$$

The optimal value of Eq. (8) is computed as $\overrightarrow{\mathbf{z}}^* = V\overrightarrow{\mathbf{w}}^*$, producing a predicted value for each example in X_f.

The approach so far explained is the general configuration of our TL algorithm for people tracking in video, thereafter we propose to use this algorithm with two different iterative schemes: *single frame transduction* and *multiple frame transduction*. A graphical representation of the two proposed schemes is given in Fig. 4.

Single frame transduction

The first possible configuration we propose is based on a single frame transduction where the target is searched among the set of people detected only in the current frame. The input of the algorithm is thus constituted by the two models X^+ and X^- of positive and negative labeled elements and the unlabeled elements X^U described by the patches F_i extracted from the considered frame i.
Labels for unlabeled elements are computed solving the QEP of Eq. (9) and we set to 0 the threshold for the class assignment. Ideally, only one label should be positive corresponding to the target whilst all the others should be negative. Due to noise of acquisition in real environments and to similarity between people appearances, multiple elements could have a positive predicted label value, we assume, with a good approximation, that the label with the highest value corresponds to the most similar element to the model of positive samples. Conversely, in case of absence of the target, i.e. miss detection or real occlusion, all the returned labels should have a value less than 0, however the TL could return a positive value and a wrong example with similar appearance could be selected. In this case the update strategy helps avoiding the propagation of the error into the model of positive samples.

Multiple frame transduction

The second setup employs a multiple frame iterative scheme. The same transductive learning algorithm can be used over the samples of people extracted in multiple frames $F_i^1, \ldots, F_i^m$. Working with these settings the threshold for the predicted label values for unlabeled elements is again fixed to 0 but the expected number of elements above this threshold is upper-bounded by the number of processed frames, i.e. the target is detected in all the frames.

Having defined both the descriptor and the detector, the complete tracking algorithm is depicted in Alg. 1.

IV. Manifold Learning Connection

The exploitation of covariance matrix as a descriptor of people images carries the additional advantage of directly connecting both the matching algorithm and the employed feature descriptors to the problem of learning the structure on a Riemannian manifold. More precisely, the laplacian of a graph is analogous to the Laplace-Beltrami operator on manifolds, eventually in our setting the Riemannian manifolds where the points expressed by covariance matrices lie onto. For every smooth, compact, m-dimensional Riemannian manifold $\mathcal{M}$ embedded in $\mathbb{R}^k$, *i.e.* a manifold for which a notion of local distance is provided, the Laplace-Beltrami operator Δ is defined for twice differentiable functions. Suppose our objective is to find a mapping $f : \mathcal{M} \to \mathbb{R}$ such that neighboring points on the manifold get mapped close together in $\mathbb{R}$. Now consider two points $\mathbf{a}$ and $\mathbf{b}$ on $\mathcal{M}$, and choose

Algorithm 1 Transductive Learning People Tracking

Require: $X^+ \longleftarrow \{x_i, y_i = +1\}$
Require: $X^- \longleftarrow \{x_i, y_i = -1\}$
while frames f **do**
  Set processing frames $F \longleftarrow \{f_j\}_{j=1}^{m}$
  people detector on F
  Compute covariance matrix for X^L and $X_{f_j}^U$ Eq. (2)
  Compute similarity matrix A_{ij}:
    $a_{ij} = \exp\left(-\frac{\rho(C_i, C_j)}{\sigma^2}\right)$
  where: $\rho(C_i, C_j)$ follows Eq. (4)
  Compute diagonal degree matrix: $Dii = \sum_j Aij$
  Compute Laplacian: $L = D - A$
  Perform Transductive Learning and compute predictions $\mathbf{z}^*$ (See Sec. III)
  Threshold $\mathbf{z}*$ and get hard class
    $X_R^+ \longleftarrow X_R^+ \cup x_i | x_i \in X^U, z_i^* \geq 0$
    $X_R^- \longleftarrow X_R^- \cup x_i | x_i \in X^U, z_i^* < 0$
  if $|X_R^+| >$ threshold **then**
    Update positive target model: $X^+ \longleftarrow X_{new}^+$
  end if
  if $|X_R^-| >$ threshold **then**
    Update negative model: $X^- \longleftarrow X_{new}^-$
  end if
end while

them sufficiently close to each other. Besides we know that for any smooth function f on a Riemannian manifold its gradient ∇f is defined as the vector such that $\langle \nabla f, T \rangle = \partial_T f$ for any tangent vector field T. If we let $c(t)$ be the geodesic curve (*i.e.* a path on the manifold) parameterized by length connecting $\mathbf{a} = c(0)$ and $\mathbf{a} = c(l)$, then we can write

$$\begin{aligned} f(\mathbf{b}) &= f(\mathbf{a}) + \int_0^l \partial_{c'(t)} f(c(t)) dt \\ &= f(\mathbf{a}) + \int_0^l \langle \nabla f(c(t)), c'(t) \rangle dt. \end{aligned} \tag{11}$$

Now by Schwartz Inequality,

$$\langle \nabla f(c(t)), c'(t) \rangle \leq \|\nabla f(c(t))\| \|c'(t)\| = \|\nabla f(c(t))\| \tag{12}$$

where the last equality holds since $c(t)$ is parameterized by length and thus $\|c'(t)\| = 1$. If we consider the first Taylor expansion of $\|\nabla f(c(t))\| = \|\nabla f(\mathbf{a})\| + o(t)$ and integrate the term in Eq. (11) over the length of the curve we have

$$|f(\mathbf{b}) - f(\mathbf{a})| \leq l \|\nabla f(\mathbf{a})\| + o(l). \tag{13}$$

If $\mathcal{M}$ is isometrically embedded (*i.e.* length of curves are preserved between the embedding space and the manifold) in $\mathbb{R}^k$ then $l = \|\mathbf{b} - \mathbf{a}\|_{\mathbb{R}^k} + o(\|\mathbf{b} - \mathbf{a}\|_{\mathbb{R}^k})$ and from Eq. (13) we obtain

$$|f(\mathbf{b}) - f(\mathbf{a})| \leq \|\nabla f(\mathbf{a})\| \|\mathbf{b} - \mathbf{a}\| + o(\|\mathbf{b} - \mathbf{a}\|). \tag{14}$$

Thus we see that $\|\nabla f\|$ provides an estimate of how far apart f maps points which are close on the manifold. Of course we want to look for a map which best preserves locality and we model this with the following optimization problem:

$$\arg\min_f \int_{\mathcal{M}} \|\nabla f(x)\|^2 dx. \tag{15}$$

Note that if we consider $\overrightarrow{\mathbf{z}}$ to be a valuable choice for f, then minimizing $\int_{\mathcal{M}} \|\nabla f(x)\|^2 dx$ is equivalent to minimizing $\overrightarrow{\mathbf{z}}^T L \overrightarrow{\mathbf{z}} = \frac{1}{2} \sum_{i,j} (z_i - z_j)^2 a_{ij}$.

V. Model Update

Using both positive and negative labeled samples as input of the transductive learning algorithm allows to have a more robust learning and a stable matching method. Of course, in order to have a powerful representation of both the target and the non-target elements, both labeled models should be iteratively updated by adding step by step new examples.

The basic idea is to exploit an update mechanism, for each model, in order to iteratively add new samples to the previous models, keeping a firm and accurate representation of the target object and avoiding the injection of classification errors. Furthermore, in this manner we can set the number of model examples limiting the number of input elements of the TL, and consequently limiting the resource consumption. Model updating is mandatory because the target appearance changes during the video, due for example to rotation, occlusions or different lighting. On the other hand, the main risk is the propagation of classification errors making the successive target assignments unreliable, eventually leading to drifting when small errors are continuously introduced at every update step.

The easiest update strategy is, of course, a *first-in/first-out* scheme where the last results are iteratively added to the models while the oldest elements are removed keeping only the elements closest in time to the current appearance of the target. A similar strategy is proposed by [60], however, despite its simplicity, this update scheme has the main drawback of having no control over errors injection. Conversely, our proposal exploits an update strategy based on clustering where the errors injection in the model is avoided by a mechanism where the insertion of new model elements do not depend on proximity in time but on the capability of the elements to represent most the observed target appearances.

Evolutionary Spectral Update

The update strategy we propose is based on *evolutionary spectral clustering*. Evolutionary clustering, [11] is a class of clustering whose aim is to process continuously evolving and time-stamped data. The problem of update the labeled elements can be considered as a dynamic task: the labeled models should represent the evolving differences in the appearance of the target and should preserve the time consistency with the previous target examples.

The target model update strategy assumes that the target examples, composing the model, can be clustered in a number of sets, each one representing a different appearance of the target, *e.g.* a different pose. Starting from the tracking results in the initial frames, we update the model adding the new elements but imposing that the initial clusters deviates smoothly from

Fig. 5. Example of a sequence of updates of the positive labeled model X^+. It can be observed how the latest model considers both past appearances and new poses of the target by mixing the target patches at different scales and positions. At time k initial elements, from time 1 are still considered into the model but a richer description of the target manifold is given exploiting more different target patches.

Algorithm 2 Evolutionary Update for positive Model X^+

Require:
$X^+_{t-1} \longleftarrow$ model at the previous iteration
$X^+_t \longleftarrow$ model at the current iteration
$X^+_R \longleftarrow$ last k results
$n \longleftarrow$ number of model elements

Define $Q \longleftarrow X^+_t \cup X^+_R$ the elements to be clustered
Compute the dissimilarity representations $\bar{Q}_t$ of elements in Q using elements in X^+_t as the basis and Eq. (4) as distance function
Compute the dissimilarity representations $\bar{Q}_{t-1}$ of elements in Q using elements in X^+_{t-1} as the basis and Eq. (4) as distance function
Compute the affinity matrices $\bar{A}_t$ and $\bar{A}_{t-1}$ respectively among the new data $\bar{Q}_t$ and $\bar{Q}_{t-1}$
Compute diagonal degree matrix: $\bar{D}_{ii} = \sum_j \bar{A}_{ij}$
Compute Evolutionary Cost EC from Eq. (16)
Perform Spectral Clustering on EC and get clusters $C_1, \ldots, C_k$
repeat
  for i=1…k **do**
    random sample x^+ from C_i
    $X^+_{new} = X^+_{new} \cup x^+$
  end for
until $|X^+_{new}| \geq n$
  return X^+_{new}

the most recent history. In this way the target appearance changes are modeled as a temporally smooth process, as shown in Fig. 5. Conversely, the negative model considers the clusters as composed by the different images of adverse targets (e.g. in the case of people tracking by all the other people images).

Following the idea proposed by [12] we decide to use an evolutionary version of the spectral clustering algorithm where the number of clusters is derived exploiting the properties of the graph laplacian. A general cost function is defined to measure the quality of the clustering results on evolving data points, this function embodies two costs. The first cost, called *snapshot cost (CS)*, measures the quality of the current clustering result w.r.t. the current data features, while the second cost, *temporal cost (CT)*, measures the temporal smoothness in terms of the goodness-of-fit of the current clustering result w.r.t. data features history. The overall cost function is thus defined as:

$$\begin{aligned} EC &= \alpha CS + (1-\alpha)CT \\ &= \alpha(\bar{D}_t - \bar{A}_t) + (1-\alpha)(\bar{D}_{t-1} - \bar{A}_{t-1}) \end{aligned} \tag{16}$$

Focusing on the spectral clustering algorithm, if the number of nodes to be clustered does not change, EC can be interpreted as a linear combination of two laplacians. In our case setting the number of nodes is equivalent to setting the number of the considered people patches (*i.e.* the last k results plus the current model elements). Referring to Eq. (16) $\bar{A}_t$ and $\bar{A}_{t-1}$ are the affinity matrices built over the model elements and the last k retrieved results respectively at time intervals t and $t-1$ (multiples of k), and $\bar{D}_t$ and $\bar{D}_{t-1}$ are the corresponding degree matrices. Specifically the two matrices $\bar{A}_t$ and $\bar{A}_{t-1}$ are computed exploiting a dissimilarity space where each element is represented in a vector space by respectively the distances with the elements of the model at time t and $t-1$ (the two representation sets at current and previous time). The affinity is then computed among the current model and last retrieved elements represented with such distances. This solution turns out to be intuitive, since the historic similarity matrix, *i.e.* the similarity matrix at the previous iteration, is scaled with a coefficient α and combined with the current similarity matrix, i.e. the similarity matrix at the current iteration. The coefficient $\alpha \in [0, 1]$, controls the update rate, with $\alpha = 1$ meaning that only the current data are clustered and $\alpha = 0$ meaning that the model is not updated maintaining the previous clustering.

The evolutionary clustering technique allows to obtain a smooth evolution of initial clustering through the combination of temporal and snapshot cost. Nevertheless, Eq. (16) outputs a laplacian that is employed to obtain the updated model. In principle, the new model is achieved by clustering the elements through the smooth laplacian, where the number of clusters k is chosen by the eigengap analysis. Subsequently, for every cluster we iteratively sample one element at a time until the number of desired elements in the model is reached.

The complete update algorithm is summarized in Alg. 2.

VI. Experimental results

In this section we assess the performance of our proposal. We organized the evaluation in two parts: the former focuses on evaluating our proposal for people tracking in surveillance scenarios, while the latter extends the evaluation to general targets disregarding the first step of detection and using instead a random patch generator. With the second part of the experiments we aim at evaluating the robustness and the general usability of the method in other contexts, and also to measure the dependence of the learner from correct detections.

A. Experiments on people tracking on surveillance datasets

We evaluate the method proposed for people tracking collecting videos from publicly available datasets THIS, CAVIAR and 3DPes. In detail:

(a) CAVIAR

(b) THIS

Fig. 6. Examples of frames taken from CAVIAR and THIS datasets.

- **THIS** [1], shown in Fig. 6 has been introduced by [56], we used the video category *Train Station*, that includes video recorded along the platforms and underpasses of a train station, mostly representing people walking alone or in groups.
- **CAVIAR** [2] is a widely used dataset for people tracking. Sequences from this dataset, and in particular *Clips from shopping center in Portugal - Corridor view* are collected in the hallway of a shopping center and show people walking, meeting with other groups and entering or exiting shops, thus contain some occlusions.
- **3DPes** [3] has been introduced by [4]. This datasets, Fig. 7, is specifically focused on re-identification, therefore we used them to evaluate critical aspects of the proposed method, *e.g.* people occluding each other, changes in appearance and pose with respect to the camera and people leaving the scene multiple times.

Referring to Fig. 7.(a), the sequence has very long occlusions when people go behind the large pilaster. In this case classical trackers do not give the same label to the same person anymore because other similar ones are visible in the area. Fig. 7.(b) is an example of small size people which make people less distinguishable and finally Fig. 7.(c) depicts different people with variable aspects, in fact people change their dresses and are detected before in frontal and then in opposite way as well as appear and disappear from the scene often.
Each sequence taken from CAVIAR and THIS datasets consists of approximately $300-600$ frames, while sequences from 3DPes have approximatively $2000-2500$ frames, with a number of people variable from 2 to 5. We tested our system using all possible target in each sequence.

[1]http://www.openvisor.org
[2]http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1
[3]http://imagelab.ing.unimore.it/visor/3dpes.asp

(a)

(b)

(c)

Fig. 7. Examples of frames taken from videos of the 3DPeS dataset.

In the evaluation people patches are detected using a conventional HOG people detector [15], the proposed graph transduction method (Sec. III) is then used to classify unlabeled patches using both positive and negative labeled elements. In our experiments the positive model, X^+, is manually initialized (e.g. the user select the target he wants to track) while the negative model X^- is composed by the all the detector outputs except the ones related to the target. Afterwards the models are automatically updated with the strategy explained in Sec. V. The systems works iteratively on each frame or set of frames depending on the choice of Single Frame Transduction (SFT) or Multiple Frame Transduction (MFT).

1) Evaluation Measures: In order to test the proposed solution we adopt measures similar to those proposed in [31] for tracking pedestrians in sparse scenes. The rationale behind this choice is that traditional per-pixel or coverage measures are not particularly suitable for tracking by detection methods. In particular we focus on evaluating how many times a target is correctly detected and tracked during the time it appears in the scene. Since our proposal is specifically designed to be robust against drifting thanks to the evolutionary spectral update algorithm, the proposed evaluation measures aim at globally evaluating tracking results over the complete ground truth sequences, *i.e.* complete people trajectories.

We measure the algorithm performances in term of:

- Ground truth (GT): number of ground truth people sequences, i.e. trajectories.
- Mostly tracked (MT) : number of sequences that are successfully tracked for more than 80% of their duration (number of target images correctly tracked divided by the ground truth target images sequence).
- Partially tracked (PT): number of sequences that are successfully tracked at least for 20% of the ground truth frames but less than 80%.
- Mostly lost (ML): number of sequences that are successfully tracked for less than 20%.

Since object tracking can be viewed as a method which is able to recover missed detections and remove false alarms from the raw detection responses, we also provide the metrics for detection evaluation. Given n_{tp}, n_{fp}, n_{fn} denoting respectively the number of true positives, false positives and false negatives, precision and recall are defined as:

- Precision (P) : the number of correctly matched detections divided by the number of output detections, $P = \frac{n_{tp}}{(n_{tp} + n_{fp})}$
- Recall (R): the number of correctly detected elements divided by the ground truth elements number, $R = \frac{n_{tp}}{(n_{tp} + n_{fn})}$

In our settings a detection is considered correct if the overlap criterion is met. The overlap criterion, used in [49], [19] assume that given O^i the object bounding box and GT^i the ground truth bounding box in frame i the overlap is computed as:

$$\frac{|O^i \cap GT^i|}{|O^i \cup GT^i|} \geq 0.5 \tag{17}$$

When Eq. (17) is satisfied, the detection is considered to match with the ground truth.

Comparison with Transductive Learning Trackers

We first assessed our complete method with single frame processing on different videos measuring precision and recall. The optimal parameters were heuristically selected in order to minimize errors in the classification and in the model update. As a result we set $\sigma = 0.4$ for the computation of the affinity matrix, $\alpha = 0.8$ for the evolutionary spectral update. We set the maximum number of elements in the positive and negative labeled models respectively as $n^+_{max} = 10$ and $n^-_{max} = 15$.

To demonstrate the effective improvement given by the conjunctive use of the positive and negative labeled elements and evolutionary update we compared our proposal (**SGT_EVO**) with two methods that use transductive learning for track assignment but employ different update schemes. In [14] the model is updated with a strategy based on the spectral properties of the graph laplacian while in [60] the weights of the labeled elements are simply decreased in time. Furthermore, to evaluate the impact of negative labels in our proposal we also evaluate the performance of our tracker using only the positive model only (**SGT_EVO$^+$**). The results obtained on the different datasets are reported in Tab. I, and show an increase in performance with respect to the previous methods. The gap in precision and recall demonstrates the effectiveness of our update strategy in modeling the changes in appearance and maintain an up-to-date representation of the target. Particularly the results display how the approach presented in this paper outperforms the other methods when working with the 3DPes videos. We would like to underline how sequences of 3DPes are more challenging when compared to videos in THIS and CAVIAR datasets. In fact people, move unpredictably changing their pose with respect to the camera, light conditions are different from point to point of the scene etc., and there is a high number of occlusions. Reaching an expressive gap in performances with these videos, thus, demonstrate the higher reliability and robustness of our method. Moreover, the same observation holds when considering the adoption of negative labels. In, particular negative labels has been observed beneficial when several people share a similar appearance but very few limited different details. In this case, the negative model helps in disambiguating the target from similar objects and brings an improvement in general performances and in particular in the recall of the system.

TABLE I
RESULTS USING DIFFERENT MODELS. VALUES ARE IN PERCENTAGE.

	THIS Dataset					
	GT	MT	PT	ML	P	R
SGT_EVO	48	**94**	**3.8**	**2.2**	**97**	**95**
SGT_EVO$^+$	48	91	6.8	2.2	94	93
[14]	48	92	5.8	2.2	91	95
[60]	48	76	15.1	8.9	92	76
	CAVIAR Dataset					
SGT_EVO	140	**92**	**7.3**	**0.7**	**96**	**94**
SGT_EVO$^+$	140	84	12	4	90	84
[14]	140	82	14.6	3.4	87	89
[60]	140	72.5	18.4	9.1	91	73
	3DPeS Dataset					
SGT_EVO	50	**58.3**	38.7	**3**	**76**	**60**
SGT_EVO$^+$	50	52	35.7	12.3	65	40
[14]	50	51.2	**32.4**	16.4	39	54
[60]	50	44.1	34.5	21.4	35	49

Qualitatively, as an evidence of the robustness of the proposed update method we selected a sequence where a strong appearance change occurs, *i.e.* sequences where people rapidly turn on themselves or remove and wear their jacket, Fig. 8. It can be noticed how, despite the complex appearance changes and some occlusions with other people in the scene, the method is able to correctly detect and follow the target while the other transductive methods failed.

TABLE II
PRECISION AND RECALL VALUES ON TEST DATASETS USING SINGLE OR MULTIPLE FRAME PROCESSING.

	Avg Prec (%)			**Avg Rec (%)**		
	THIS	CAVIAR	3DPeS	THIS	CAVIAR	3DPeS
SFT	**97**	96	76	95	**94**	60
MFT 3	96	96	84	**96**	93	**66**
MFT 5	94	93	**85**	94	92	53
MFT 8	95	**97**	79	90	**94**	42

Fig. 8. 1st and 2nd rows: Some example frames of a sequence specifically focused on testing robustness in case of appearance changes. 3rd row: Obtained results.

Evaluation of Multiple Frame Processing

We further evaluate the impact on precision and recall of the proposed solution when processing either a single frame (SFT) or n frames with $n = 1, 3, 5, 8$ (MFT n). As shown in Table II we obtained comparable results on all the different configurations meaning that the system has a noticeable degree of robustness in terms of the number of unlabeled samples provided as input. A slight improvement in both precision and recall can be seen when using a number of 3 frames, whilst a decrease is obtained processing an increased number of frames. This can be reasonably explained because when the number of input frames increases beyond a certain value the appearance of the target may change non-smoothly with respect to the model. Consequently the unlabelled images relations exhibit inconsistencies that negatively affect the transduction process. At this level, the benefit of having a greater number of unlabelled data, keeping the labelled amount fixed, is dismissed by the larger variability of input elements that increases the complexity of the TL graph partitioning problem.

Evaluation of the model update strategy

In order to test the robustness against drifting of the model update strategy we performed some experiments by manually injecting wrong patches in the tracking process. The errors have been injected both in the model and in the last results and we test the capability of the update strategy to build a new model without errors. The plot in Fig. 9 shows the precision of the updated model referred to a varying number of the wrong patches considered. In the experiments we consider a number $n = 10$ of elements in each model and a number $k = 6$ of tracking results (which means the update is performed every 6 frames). The x-axis of the chart reports the number of wrong patches in the last k results. Assuming that the update strategy might fail and errors might be injected in the current model, we also considered a different number of wrong patches in the current model (represented in the chart by the different lines and $M.Err$ value).

The results show the capability of the evolutionary update strategy to reject errors and avoid drifting, especially when the labeled model is not contaminated by errors. When errors are injected in the current model, the update strategy allows to recovery from the drifting when the number of good patches is not too low. This behavior is motivated by the use of evolutionary clustering in the update process. The evolutionary clustering impose the clusters in the model change smoothly in time. Errors do not represent a smooth change and are absorbed by correct clusters being subsequently rejected because far from clusters centers.

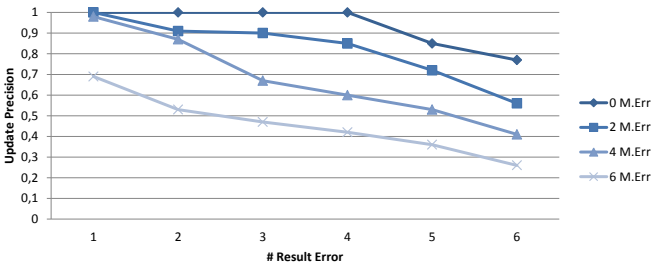


Fig. 9. Evaluation of the evolutionary spectral update. Errors are injected in the model and in the last results. The different curves consider an initial target model with a variable number of errors inside (M.Err from 0 to 6 out of 10 images in the model). Every curve evaluates the tracking precision when an increasing number of wrong results are considered as positive target images (i.e. when the tracker make a subsequent set of wrong assignments) and hence used for updating the target model. The x-axis reports the varying number of errors considered in the last results (with a total number of 6 results considered). The different lines report the performances when the current model is contaminated with a different number of wrong patches.

Comparisons on CAVIAR dataset

Finally, on the CAVIAR dataset we compare our proposal with several learning based tracking by detection approaches in

TABLE III
PERFORMANCE COMPARISON ON CAVIAR DATASET. VALUES ARE IN PERCENTAGE.

	GT	MT	PT	ML	P	R
[58]	140	84.3	12.1	3.6		81.8
[24]	143	78.3	14.7	7.0		86.3
[28]	140	82.3	10	7.7	70	93.8
[22]	143	85.3	13.7	1.0	80	**96.3**
[36]	143	84.6	14.0	1.4		89.0
[31]	143	84.6	14.7	**0.7**	**96.9**	89.4
SGT_EVO	140	**92**	**7.3**	**0.7**	96.0	94.0

order to show the performance improvement using transductive learning and spectral updating for people tracking in typical surveillance scenarios. The CAVIAR dataset contains 26 video sequences of a corridor in a shopping center taken by a single camera with frame size of 384×288 and frame rate of 25fps. As in previous tests the detection of the people in the scene is obtained using the HOG based people detector [15] and transductive people tracking is run over the detected snapshots in every frame. Table III shows the result assessed by our proposal in comparison with different learning based tracking approaches.

We compared our method against methods that perform data association among small fragments of reliable tracks, *tracklets*, and then perform a data association stage among them to recover full tracks [58], [24]. The performance improvement over these approaches is due to the fact that our proposal performs data association among people patches independently frame by frame, leading to a more effective recovery in case of errors. Additionally we compare our proposal with two particle filtering multiple target tracking approaches, [22], [28]. The former method employs a discriminative pre-training stage to improve particle filter responses by finding the best set of hyperparameters, while the latter add a joint term to the filter probability that encodes the joint influence among targets. Nevertheless, the improvement over particle filter method is due to the fact that when target confusion occurs in the scene (e.g. targets are visually similar and/or follow similar paths) particle filters tend to produce wrong assignments. The smoothing factor introduced by modeling the target states through probability distribution mitigates the errors but also averages the small discriminative details of target themselves. Finally, we compared our solution with two model learning solutions, [31], [36], that exploit boosting to select and rank the best tracked candidates in subsequent frames with an approach similar to our proposal. Even in these cases, we are able to obtain a slight improvement, thanks to the transductive learning algorithm, that takes advantage also from the relation among unlabeled and not only from labeled data. Conversely from [31], [22], we do not make use of any assumption about objects motion obtaining a more general solution applicable even in case of abrupt motion or strong occlusions.

Eventually, we tested the robustness of the tracker w.r.t. the detector performances on the CAVIAR dataset. Precisely, we randomly injected detection errors in the detector results varying the precision and recall according to the curve depicted

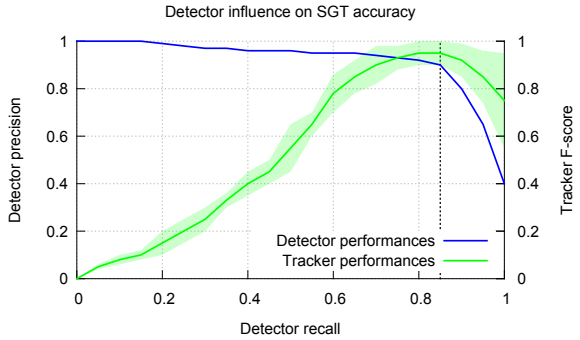


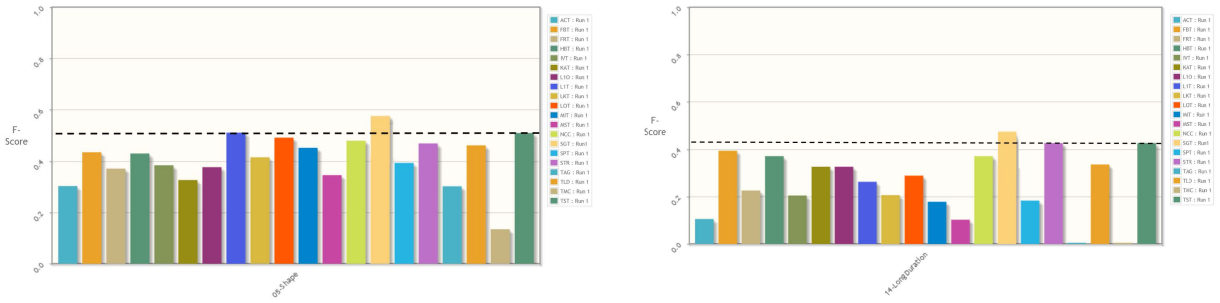
Fig. 10. Evaluation of SGT performance (green curve) varying the detector precision and recall (blue curve). The dashed line intercepts the tracker performance curve at its maximum value.

in Fig. 10. Accordingly, we evaluated the tracker results in terms of F-score using the different generated inputs. We performed several runs of both the detection and tracking processes and the shaded area in the figure shows the tracker score variance. The results underline the dependence of the tracker on the detector recall. The transductive learning process is able to reject false detections exhibiting a robustness against precision drops. On the contrary, due to its discriminative nature, SGT is not able to cope with the situations where the recall is too low; the tracker does not generate new target candidates and consequently if the correct candidate is not present in the input set it is obviously not tracked. Under this premises, experiments in Sec. VI-B, evaluates the SGT performance in a sliding-window like scenario where the input patches are sampled around the last target position.

B. Experiments on general targets and camera settings

This last part is devoted to the evaluation of the suitability of the proposal to follow a general target without any a priori assumption on both the type of camera and the acquired video. In the experiment, we replaced the detector with a patch generator and we evaluate the proposal on the **ALOV** a recent state of the art benchmark for target tracking, [49]. This dataset has been specifically proposed to cover the various situations that may happen in real-life videos. The 315 videos are focused on different aspects such as illuminations, transparency, specularity, confusion with similar objects, clutter, occlusion, zoom, severe shape changes, different motion patterns, low contrast, etc. To better understand the challenges of the dataset, some visual results are depicted in Fig. 11.

The positive and negative model are initialized respectively with patches generated by uniform sampling around the given initial target position, and with patches taken outside of a fixed radius from the initial target. During tracking, positive patches are generated uniformly within a fixed radius around the last target position while negative patches are generated randomly outside the positive patches radius. The number np of patches is equal for positive and negative patches (in our experiments we set $np = 10$). Additionally, to account for scale changes patches are generated at different scale considering smooth changes limited by twice the size of the initial target.



(a) (b)

Fig. 12. F-measure of the proposed method SGT in the shape and long duration categories of the ALOV dataset. The black dashed line is the top performance reported in [49] from state of the art trackers.

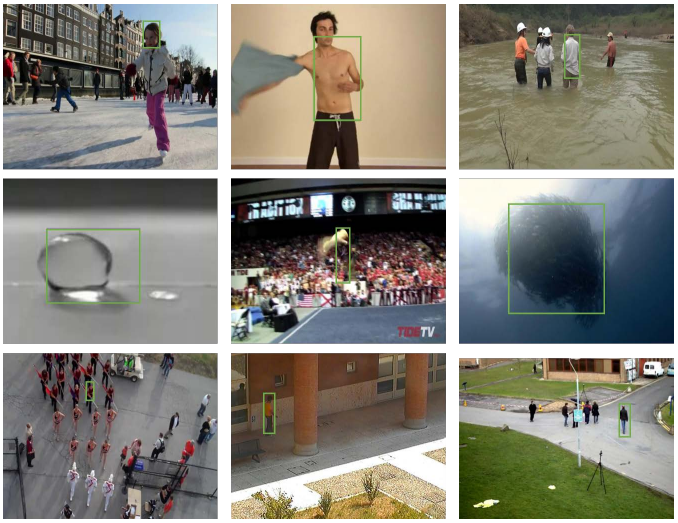


Fig. 11. Visual results of the proposed method on the ALOV dataset of [49].

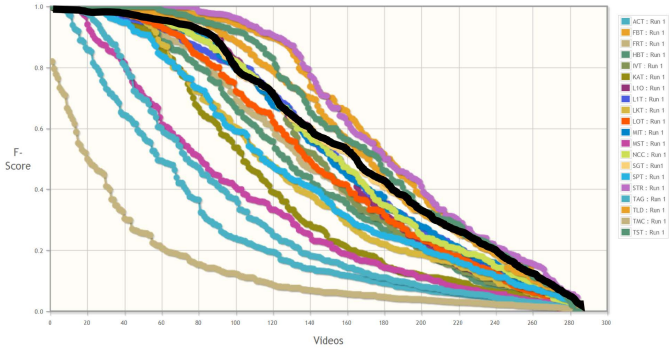


Fig. 13. The survivor curve on the ALOV dataset. Results are compared with the state of the art tracker results reported in [49]. Black curve are the results of the proposed method, namely SGT.

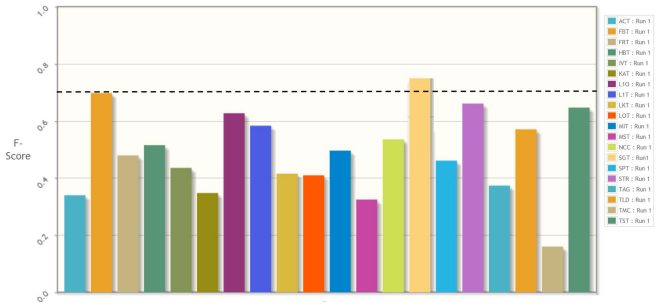


Fig. 14. F-measure of the proposed method SGT on the subset of ALOV dataset videos that contain people as a whole figure. Results are compared with the tracker evaluated in [49] and the dashed black line reports the top performance of state of the art trackers.

Evaluation Measures: In these experiments, according with [49], the performance are evaluated in terms of F-measure:

$$\text{F-measure} = 2 \cdot \frac{P \cdot R}{P + R} \tag{18}$$

where P and R are respectively precision and recall. Since the patches are randomly generated, and not given by a detector, the overlap between the considered bounding box and the ground truth is computed in order to decide whether the track matches or not the ground truth. We use the overlap criterion of Eq. (17) to identify whether a track is considered to match with the ground truth. The obtained results are compared against the results published in [49] on the set of tracker selected in the experimental survey. In Fig. 13, analogously with the evaluation in [49], we plot the survivor curve of our proposal and compare against the ones evaluated in the experimental survey. The black curve refers to our complete proposal and, despite not specifically designed for general target tracking, it is competitive w.r.t the state of the art. Some remarkable results that evidence the effectiveness of our model update strategy can be observed by the individual results on the specific categories, respectively shape category and long duration, Fig. 12(a-b). In the former the tracked target is subject to smooth shape transformation (e.g. people bending or waving hands), in the latter the aim is to track the target for a significantly long amount of time during which different changes in the target appearance may occur (e.g. camera zooms, light changes, different viewpoints and target poses ...).

Finally, the tracker have been tested on a subset of videos where people have been recorded as a whole figure. This test is closer to the design hypothesis of our proposal. In particular, our method was designed to track people in surveillance scenarios and consequently the adopted features are particularly suitable to resolve this task. In Fig 14 we can observe that our method outperforms, in term of F-measure, the tested trackers in [49].

VII. Conclusions

In this paper we proposed a method for people and object tracking based on a transductive learning algorithm. The learning process exploits both positive and negative labeled elements and mimic the process of learning the manifold structure that represents the tracked target. Moreover this connection is achieved by selecting the covariance matrix as the representative feature of the target; in this manner it is possible to exploit the fact that covariance matrices are points on a Riemannian manifold and their distance is expressed in terms of geodesic distance making the learning step consistent with a manifold learning framework. To limit the drift, that occurs by updating the model with wrong results, a smooth model update procedure has been designed that updates the model by means of evolutionary clustering enforcing temporal smoothness in the model elements. The experiments has evidenced the advantage of using both positive and negative element in the model and, compared to different tracking solution, our method have exhibit competitive performances when the target are people that possibly change their appearance (e.g. by turning around, or removing jackets...). Moreover the proposal has been tested to general objects with satisfactory results, outperforming state of the art methods in the scenarios where people is present evidencing that the conjunction of covariance matrices and manifold learning are a robust method for people tracking.

Acknowledgments

This work was carried out within the project *La Cittá Educante* of the National Technological Cluster on Smart Communities funded by the Italian Ministry of Education, University and Research, MIUR.

References

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

[2] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 983–990, June 2009.

[3] S. Bak, E. Corvee, F. Bremond, and M. Thonnat. Person re-identification using spatial covariance regions of human body parts. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 435–440, September 2010.

[4] D. Baltieri, R. Vezzani, and R. Cucchiara. 3dpes: 3d people dataset for surveillance and forensics. In *Proceedings of International ACM Workshop on Multimedia access to 3D Human Objects*, November 2011.

[5] J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 744–750, June 2006.

[6] D. Borghesani, D. Coppi, C. Grana, S. Calderara, and Cucchiara R. Feature space warping relevance feedback with transductive learning. In *Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 70–81, August 2011.

[7] J.E. Boyd and J. Meloche. Evaluation of statistical and multiple-hypothesis tracking for video traffic surveillance. *Machine Vision and Applications*, 13(5-6):344–351, March 2003.

[8] Y. Cai, N. de Freitas, and J.J. Little. Robust visual tracking for multiple targets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 107–118, May 2006.

[9] S. Calderara, R. Cucchiara, and A. Prati. Bayesian-competitive consistent labeling for people surveillance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):354–360, February 2008.

[10] L. Cehovin, M. Kristan, and A. Leonardis. Robust visual tracking using an adaptive coupled-layer visual model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):941–953, April 2013.

[11] D. Chakrabarti, R. Kumar, and A. Tomkins. Evolutionary clustering. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, pages 554–560, August 2006.

[12] Y. Chi, X. Song, D. Zhou, K. Hino, and B.L. Tseng. On evolutionary spectral clustering. *ACM Transaction on Knowledge Discovery from Data*, 3(17):1–30, December 2009.

[13] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 142–149, June 2000.

[14] D. Coppi, S. Calderara, and R. Cucchiara. Appearance tracking by transduction in surveillance scenarios. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 142–147, September 2011.

[15] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, June 2005.

[16] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):743–761, February 2011.

[17] G. Doretto, T. Sebastian, P. Tu, and J. Rittscher. Appearance-based person reidentification in camera networks: problem overview and current approaches. *Journal of Ambient Intelligence and Humanized Computing*, 2(2):127–151, June 2011.

[18] A. Ess, B. Leibe, K. Schindler, and L. van Gool. Robust multiperson tracking from a mobile platform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10):1831–1846, October 2009.

[19] M. Everingham, L. van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[20] N. Faraji Davar, T. de Campos, J. Kittler, and F. Yan. Transductive transfer learning for action recognition in tennis games. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1548–1553, November 2011.

[21] W. Forstner, B. Boudewijn Moonen, and C.F. Gauss. A metric for covariance matrices. Technical report, Department of Geodesy and Geoinformation, Vienna University of Technology, 1999.

[22] R. Hess and A. Fern. Discriminatively trained particle filters for complex multi-object tracking. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 240–247, June 2009.

[23] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):663–671, April 2006.

[24] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 788–801, October 2008.

[25] Y. Huang, Q. Liu, S. Zhang, and D.N. Metaxas. Image retrieval via probabilistic hypergraph ranking. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3376–3383, June 2010.

[26] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1409–1422, May 2012.

[27] R. Kaucic, A.G.A. Perera, G. Brooksby, J. Kaufhold, and A. Hoogs. A unified framework for tracking through occlusions and across sensor gaps. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 990–997, June 2005.

[28] Z. Khan, T. Balch, and F. Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1805–1819, November 2005.

[29] Z.W. Kim. Real time object tracking based on dynamic feature grouping with background subtraction. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

[30] H. Kjellstro, D. Kragic, and M.J. Black. Tracking people interacting with objects. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 747–754, June 2010.

[31] CH. Kuo, C. Huang, and R. Nevatia. Multi-target tracking by on-line learned discriminative appearance models. In *Proceedings of the IEEE*

International Conference on Computer Vision and Pattern Recognition (CVPR), pages 685–692, June 2010.

[32] J. Kwon and K.M. Lee. Tracking by sampling trackers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1195–1202, 2011.

[33] Y. Lao, J. Zhu, and Y.F. Zheng. Sequential particle generation for visual tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(9):1365–1378, September 2009.

[34] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1683–1698, oct. 2008.

[35] F. Li and H. Wechsler. Open set face recognition using transduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1686–1697, November 2005.

[36] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2953–2960, June 2009.

[37] T. Lin and H. Zha. Riemannian manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):796–809, March 2008.

[38] W. Liu and S.F. Chang. Robust multi-class transductive learning with graphs. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 381–388, June 2009.

[39] E. Maggio, M. Taj, and A. Cavallaro. Efficient multitarget visual tracking using random finite sets. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(8):1016–1027, August 2008.

[40] L. Matthews, T. Ishikawa, and S. Baker. The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):810–815, June 2004.

[41] G. Meng, L. Huaping, and S. Fuchun. Visual tracking using online semi-supervised learning. In M. Kamel and A. Campilho, editors, *Image Analysis and Recognition*, Lecture Notes in Computer Science, pages 406–415. Springer Berlin / Heidelberg, 2011.

[42] M.J. Metternich, M. Worring, and A.W.M. Smeulders. Color based tracing in real-life surveillance data. In Yun Q. Shi, editor, *Transactions on Data Hiding and Multimedia Security V*, Lecture Notes in Computer Science, pages 18–33. Springer Berlin / Heidelberg, 2010.

[43] H.T. Nguyen and A.W.M. Smeulders. Robust tracking using foreground-background texture discrimination. *International Journal of Computer Vision*, 69(3):277–293, September 2006.

[44] K. Okuma, A. Taleghani, N. De Freitas, O. De Freitas, J.J. Little, and D.G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 28–39, May 2004.

[45] A.G.A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and W. Hu. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 666–673, June 2006.

[46] F. Pernici and A. Del Bimbo. Object tracking by oversampling local features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12):2538–2551, October 2014.

[47] F. Porikli, O. Tuzel, and P. Meer. Covariance tracking using model update based on lie algebra. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 728–735, June 2005.

[48] D.A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1-3):125–141, May 2008.

[49] A. Smeulder, D. Chu, R. Cucchiara, S. Calderara, A. Deghan, and M. Shah. Visual tracking: an experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1442–1468, July 2014.

[50] S. Stalder, H. Grabner, and L. Van Gool. Beyond semi-supervised tracking: Tracking should be as simple as detection, but not simpler than recognition. In *Proceedings of the International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1409–1416, September 2009.

[51] D. Svensson, M. Ulmke, and L. Danielsson. Joint probabilistic data association filter for partially unresolved target groups. In *Proceedings of the Conference on Information Fusion (FUSION)*, pages 1–8, July 2010.

[52] F. Tang, S. Brennan, Q. Zhao, and H. Tao. Co-tracking using semi-supervised support vector machines. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, October 2007.

[53] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Review*, 43(2):235–286, 2001.

[54] V. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, September 1998.

[55] J. Vermaak, A. Doucet, and P. Pérez. Maintaining multi-modality through mixture tracking. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1110–1116, October 2003.

[56] R. Vezzani and R. Cucchiara. Video surveillance online repository (visor): an integrated framework. *Multimedia Tools and Applications*, 50(2):359–380, November 2010.

[57] S. Wang, H. Lu, F. Yang, and M.-H. Yang. Superpixel tracking. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1323–1330, November 2011.

[58] J. Xing, H. Ai, and S. Lao. Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1200–1207, June 2009.

[59] W. Ying and T.S. Huang. Color tracking by transductive learning. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 133–138, June 2000.

[60] Y. Zha, Y. Yang, and D. Bi. Graph-based transductive learning for robust visual tracking. *Pattern Recognition*, 43(1):187–196, January 2010.

[61] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

[62] T. Zhao and R. Nevatia. Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1208–1221, September 2004.

Dalia Coppi received PhD degree in 2014 from the University of Modena and Reggio Emilia. Her current research interests include tracking and human computer interaction.

Simone Calderara received the PhD degree in ICT in 2009 from the University of Modena and Reggio Emilia, where he is now an assistant professor within the Imagelab group. His current research interests include computer vision and machine learning applied to human behavior analysis, visual tracking in crowded scenarios and time series analysis for forensic applications.

Rita Cucchiara received her master degree in electronic engineering and the PhD degree in computer engineering from the University of Bologna, Italy, in 1989 and 1992 respectively. Since 2005, she is a full professor at University of Modena and Reggio Emilia, Italy, where she heads the ImageLab group and the SOFTECH-ICT research center. Her research focuses on pattern recognition, computer vision and multimedia.