Appearance tracking by transduction in surveillance scenarios

Dalia Coppi, Simone Calderara, Rita Cucchiara
DII - University of Modena and Reggio Emilia
Via Vignolese 905 - Modena - Italy
{name.surname}@unimore.it

Abstract

We propose a formulation of people tracking problem as a Transductive Learning (TL) problem. TL is an effective semi-supervised learning technique by which many classification problems have been recently reinterpreted as learning labels from incomplete datasets. In our proposal the joint exploitation of spectral graph theory and Riemannian manifold learning tools leads to the formulation of a robust approach for appearance based tracking in Video Surveillance scenarios. The key advantage of the presented method is a continuously updated model of the tracked target, used in the TL process, that allows to on-line learn the target visual appearance and consequently to improve the tracker accuracy. Experiments on public datasets show an encouraging advancement over alternative state-of the-art techniques.

1. Introduction

The problem of object tracking, especially people, has been a focus in the last years for which many different approaches have been developed. A comprehensive review and classification of published literature is presented in [18]. Despite the wide range of existing methods, most of the state-of-the art proposals are centred on Particle Filtering (PF) tracking [12, 2, 9, 4]. PF has been proven to be a robust algorithm with the underlining idea to represent the tracks association of the posterior density function by a set of random samples with associated weights to compute estimates based on these samples and weights. Despite the effectiveness of this solution the difficulty of the model representing objects under varying and different lighting conditions, leads to inaccurate results in complex scenarios. Instead, a completely different perspective is to solve the tracking problem as a graph partitioning problem, where the objective is to find connected paths that link nodes belonging to the same target. This approach has been proficiently applied in computational video forensic [11], where people are searched in videos by their appearance using a similarity graph. The proposal sounds solid and cogent but not suitable for on-line video processing and relies on the hypothesis of selecting both the first and the last target image. Recently, graph partitioning has been adopted in people video tracking [20], where the appearance matching problem is solved using a semi-supervised learning approach. The drawback of this method is its heavy reliance on the quality of training data. The tracker accuracy tends to reduce if errors are injected in the target model.
In this paper, similar to [20], we propose to exploit the Transductive Learning algorithm in conjunction with a Riemannian similarity measure iteratively learning, frame-by-frame, the Riemannian manifold that represents the target appearance. This tracking interpretation allows to neglect the way snapshots are extracted from the video frames and to relax the constraint on the type of camera, fixed or PTZ. The most interesting aspect of our approach is the model construction, where a set of images continuously updated, is used instead of an aggregated statistical representation of them, allowing in a straightforward way to deal with wide changes in the target appearance, occlusions and reacquisition. A second novelty that we introduce, is the use of spectral properties of the eigenvalues combined with the Transductive Learning algorithm to obtain a model integrity check that helps to avoid errors and their propagation.

2. Transductive Inference for Target Tracking

We propose to explore the problem of tracking a target using its appearance from a classification-inspired perspective by learning about the target model and searching for its realization in the video sequences. The learning process is continuously updated in time, in order to improve the classification accuracy, as long as the system is being provided with new examples of the target. Considering the problem of having a set of n possible target candidates for every frame extracted by an object detection algorithm(e.g. people detection, motion segmentation ...) $X = \{x_i | i = 1 \dots n\}$ and supposing to have a model of the desired target composed by a set of k examples $X_M = \{m_i\} | i = 1 \dots k\}$. The complete dataset comprises both, the model and the

978-1-4577-0845-9/11/$26.00

candidates' samples with their associated label function y_i, which takes positive values for the element belonging to the model.

$$D(X,Y) = \{X \cup X_M, Y : y_i = 1 \, iff \, x_i \in X_M\} \quad (1)$$

Setting the problem in this form, we aim for each frame, to propagate the knowledge encoded into the model that can be equivalently interpreted as the problem of *estimate the missing label function values given the model ones*, this recalls the settings of the transductive inference problem formulation given by Vapnik, [14]. A possible solution to this problem can be obtained by minimizing the leave-one-out (loo) error on classification, using a trivial K-Nearest-Neighbour rule, where an error occurs when the majority of the neighbours of elements (x_i, y_i) are not from the same class. Recalling [8] we can set an upper bound to loo error of the classifier:

$$Err_{loo}^{knn}(X,Y) \leq \sum_{i=1}^{N}(1-\delta_i) \quad (2)$$

where δ_i is the KNN margin $\delta_i = y_i \frac{\sum_{j \in KNN(x_i)} y_j w_{i,j}}{\sum_{m \in KNN(x_i)} w_{i,m}}$ and $w_{i,j}$ is the similarity between x_i and x_j. The minimization of Eq. 2 can be obtained by maximizing the margin δ_i, imposing constrained values on the model labels:

$$\begin{aligned} &\max_y y^T A y \text{ s.t.} \\ &y_i = \pm 1 \text{ if } x_i \in X_M \\ &y_j \in \{0,1\} \end{aligned} \quad (3)$$

and

$$A'_{i,j} = \begin{cases} \frac{w_{i,j}}{\sum_{k \in KNN(x_i)} w_{i,k}} \text{ if } x_j \in KNN(x_i) \\ 0 \text{ otherwise} \end{cases} \quad (4)$$

In particular, considering A as the adjacency matrix of a graph G where nodes represent the samples and edges the similarity between samples, Eq. 3 can be solved using the s-t mincut algorithm finding the cut $cut(G^+, G^-)$ of the graph (where G^+ and G^- denote the set of vertices representing, respectively, positive and negative samples) that minimizes the quantity $2 * cut(G^+, G^-)$ or,equivalently, maximizing Eq. 3. This solution is connected with the transductive learner of Blum and Chawla [1], following the intuition that the separation is between positive and negative examples and puts strongly connected elements into the same class. Although the solution is elegant and appealing, it easily leads to degenerate cuts, if the number of labeled samples is not well balanced. This can often occur in surveillance settings when the number of elements extracted in the current frame strongly differs from the number of examples in the model. To overcome unbalanced cuts, which have been demonstrated occurring also in transductive SVM, due to the constraint of fixing a-priori the number of positive samples in the test set [8], a possible solution is to include the cut size in the objective function leading to the ratio-cut problem:

$$\begin{aligned} &\max_y \frac{cut(G^+, G^-)}{|i : y_i = 1||i : y_i = 0|} \text{s.t.} \\ &y_i = 1 \text{if} x_i \in X_M \\ &y_j \in \{0,1\} \end{aligned} \quad (5)$$

The ratio-cut problem is unsupervised and a solution can be computed when no constraints are given. The constraint on y makes the problem semi-supervised; however letting y assume real values and exploiting spectral properties of the graph Laplacian leads to an efficient way in finding a solution to the balanced ratio-cut problem in a semi-supervised way. Graph Laplacians have been successfully adopted in recent image segmentation [5], spectral clustering and dimensionality reduction [10], since they represent a powerful manifold learning tool.

In our method, and generally in graph-based transductive methods, vertices of the graph represent the labeled and unlabeled samples while the edges between the samples reflect the similarity among them.

Let us restrict the case of a two class problem, where the first is the class of the samples that belong to the target model, the other one is the class of all the negative samples.

Analogously to [20] we define a continuous function $f : R^n \rightarrow [0\ 1]$, which denotes each sample confidence to belong to one class. This function is the relaxed version of y and can assume continuous values allowing to find a solution to the ratio-cut. We can associate to f a cost function analogous to Eq. 5 $J(f)$ which is:

$$J(f) = \sum_{i,j=1}^{n} \|f(x_i) - f(x_j)\|^2 w_{ij} + \lambda \sum_{i=1}^{k} \|f(x_i) - y_i\|^2 \quad (6)$$

where $\lambda > 0$ is a regularization parameter and the quadratic penalty $\|f(x_i) - y_i\|^2$ embodies the semi-supervised constraint on model labels values. The right part of Eq. 6 encodes the local information, while the left one constrains on global information. In order to obtain the solution of Eq. 6, it is necessary to find the minimum.

$$f* = argmin \, J(f) \quad (7)$$

If we rewrite Eq. 6 in matrix notation, we obtain:

$$J(f) = (f(X) - Y)^T (f(X) - Y) + \lambda f(X)^T L f(X) \quad (8)$$

where Y is the labels vector and L is the graph Laplacian computed in its unnormalized form [10], $L = D - W$ where W is the affinity matrix, whose elements represent the similarity between samples in the data set (it will be discussed in the following) and D is the diagonal Degree matrix, whose diagonal elements are the degree of vertices:$d_{ij} = \sum_{j=1}^{n} w_{ij}$. For these reasons the constructed graph Laplacian is symmetric and positive semidefinite.
The solution of Eq. 8 is computed setting to zero the first order partial derivatives of $J(f)$ w.r.t. f, $\partial J(f)/\partial f = 0$, obtaining:

$$f* = (I - \lambda L)^{-1} Y \tag{9}$$

where $f*$ contains the class confidence of each new sample obtained propagating the model labels in a transductive way to test elements avoiding degenerated solutions due to an unbalanced number of samples in the model and in the test set.

2.1. Affinity matrix computation for video tracking

In the previous section, matrix W has been used for Laplacian matrix computation. W is the affinity graph built over positive samples and on samples belonging to the current frame. Vertices of the graph represent the samples with the edges between them reflecting the similarity between each couple of vertices. Fig. 1 is a representation of an affinity graph with the left part showing the labeled samples' nodes and the right part the unlabeled samples extracted from the current frame.

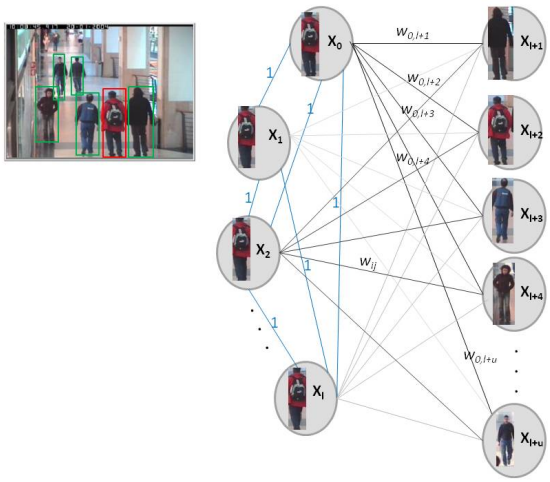

Figure 1. Example of affinity graph.

We divided W in four parts:

$$W = \begin{pmatrix} W^{ll} & W^{lu} \\ W^{ul} & W^{uu} \end{pmatrix} \tag{10}$$

where the subgraphs W^{ul} and W^{lu} are symmetrical. The graph W is therefore composed of three subgraphs:

- W^{ll} is the subgraph of labeled data (previous tracked results belonging to the target model) and it can be represented by a fully connected graph. Since elements are the previous tracked results, weights between them are constrained to a fixed value k: $w_{ij}^{ll} = k$.
- W^{uu} is the subgraph of unlabeled samples or, in other words, of the candidate patches extracted from the current frame. Since in each frame the target object can appear only once, just one of the candidate objects can belong to the positive samples, therefore this subgraph is non-connected and the similarity between unlabeled samples is constrained to 0: $w_{ij}^{uu} = 0$;
- $W^{lu} = W^{ul}$ is the subgraph between the labeled samples and the unlabeled samples. The goal is to connect samples x_i and x_j with a weight proportional to the similarity between them. This can be done weighting the edges with the affinity: $w_{ij}^{lu} = w_{ij}^{ul} = \exp\left(-\frac{\rho(x_i, x_j)}{\sigma^2}\right)$ where $\rho(x_i, x_j)$ is the distance between samples x_i and x_j and σ is a regularization parameter.

3. Samples Feature Model and Distance Computation

In order to perform Transductive Learning on people images, we need to define a proper feature model and a distance measure to capture similarities among their appearances. When adopting a graph Laplacian method we make the assumption that (input) points are generated by a probability distribution with support in a sub-manifold of the Euclidean Space. Under this assumption, we want to learn about the target model structure capturing the geometry of the manifold it lies onto, using its realization (the model images) and subsequent iteration of Transductive Learning. Among possible image descriptors covariance matrices [13], that have been effectively adopted in different contexts [17, 16], suit our purpose. They carry the advantage of being compact descriptors of an image representing points on high dimensional Riemannian manifold, with distances between them representing geodesic distances on the manifold.
Initial labeled examples constitute a first set of samples from the manifold structure and, during subsequent transductions, the structure is refined with the addition of new elements (i.e. new tracked samples). This process allows us to compute the object model and the Riemannian manifold geometry by exploiting the convergence of the graph Laplacian eigenvectors to the discrete version of the Laplace-Beltrami operator eigenfunctions in the continuous real domain [15]. Once the structure has been successfully computed and the model inferred, we can exploit the property

of invariance of the manifold descriptors to isomorphism, in conjunction with the rotation and illumination invariance of the covariance matrices in order to obtain a robust object model that can be used for both, tracking and reacquisition, under scene and object appearance changes.
The covariance matrix is a square symmetric matrix $d \times d$, with d being 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. Generally a single matrix extracted from a region is enough to match the region in different views and poses, since the noise corrupting individual samples is largely filtered out with the average filter during covariance computation. Moreover, covariance matrices can have scale and rotation invariance properties and are independent to the mean changes such as identical shifting of color values. Focusing on non-singular covariance matrices, we can observe that they are positive definite, and they can be formulated as a connected Riemannian manifold.
Let I be a three-dimensional color image and F be the $W \times H \times d$ dimensional feature image extracted from I,

$$F(x, y) = \Phi(I, x, y) \tag{11}$$

where the function Φ can be any mapping such as intensity, color, gradients, filter responses, etc. Let $\{z_i\}_{i=1...N}$ 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:

$$C_R = \frac{1}{N-1} \sum_{i=1}^{n} (z_i - \mu)(z_i - \mu)^T \tag{12}$$

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 by its spatial, color and edge information. We use x and y pixel location in the image grid, RGB color values and Gx and Gy first order derivatives of the intensities calculated through Sobel operator w.r.t. x and y. Therefore each pixel of the image is mapped to a seven-dimensional feature vector $z_i = [\ x\ \ y\ \ R\ \ G\ \ B\ \ Gx\ \ Gy\ ]^T$. Based on this features vector the covariance of a region is a 7×7 matrix.
To obtain the most similar region to the given object, we need to compute the distances between the covariance matrices corresponding to the target object and the candidate regions. However, the covariance matrices do not lie on the Euclidean space. Therefore an arithmetic subtraction of two matrices would not measure the distance of the corresponding regions. The distance metric between the covariance matrices is proposed in [7] as the sum of the squared logarithms of the generalized eigenvalues.

$$\rho(C_i, C_j) = \sqrt{\sum_{i=1}^{d} ln^2 \lambda_k(C_i, C_j)} \tag{13}$$

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

$$\lambda_k C_1 x_k - C_j x_k = 0 \ \ k = 0 \ldots d \tag{14}$$

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

4. Target model update strategy

Video tracking is a continuous process and our proposal relies on the adoption of a transductive learner to infer the Riemannian structure model from samples in order to classify new elements, whether belonging or not to the tracked target. At this aim, ad-hoc model updates strategies must be adopted to avoid the injection of wrong elements in the collection of model samples. The selection of positive samples is important for the proper functioning of the transductive learner and for this reason we update the model of positive samples at each iteration of the algorithm. The positive samples are denoted as:

$$\Delta = (x_i^p, y_i)_{i=1,\ldots,n} \tag{15}$$

with y_i being the confidence of the i-th samples. The confidence is in the range of $[0\ 1]$ and when the initial positive samples are selected by the user their confidence is set to the highest value 1. In the subsequent frames a *forgetting factor* α is used in order to decrease the confidence of the oldest positive samples, $\Delta = \alpha * \Delta$. We use this update mechanism under the hypothesis that the samples, closest in time to the current object, are more similar than the others and thus their confidence should be higher.
After all the samples' confidences are calculated through Transductive Learning and graph Laplacian, the patches with confidence below a given threshold are deleted from the positive samples. This allows to preserve a coherent positive model. To avoid that misclassified samples corrupt the model, we propose to check *the integrity of the iteratively constructed track model* exploiting the spectral properties of the graph Laplacian.
We mentioned before how Transductive Learning is performed on each frame of the video, this means that we have a tool to determine for every frame whether the considered object should be added to the track of the target person. Before adding this object definitively to the positive samples we check if the integrity of the target model is maintained.
If we consider the Laplacian matrix L built over the model

of positive detected object, it is possible to exploit the spectral properties of the graph to determine the number of clusters in which the data should be divided. Particularly we follow the method delineated in [19] according to which the Normalized Laplacian matrix, L_{sym} is used.

$$L_{sym} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \tag{16}$$

Let $\lambda_0, \lambda_1, \ldots, \lambda_k$ be the eigenvalues of L_{sym}, the number of eigenvalues $\lambda_i = 1$ is equal to the number of groups in the examined data. Since we are not operating with clean dataset but with real data, the eigenvalues start to deviate from 1 due to the introduced noise. For this reason we suggest not to count the number of eigenvalues $\lambda_i = 1$, but to use a threshold instead, empirically set to $\lambda_i > 0.9$ in our experimental settings. The threshold adjusts the accuracy of the tracker allowing to trade-off the precision of the tracked results versus the recall of the tracker. If the eigenvalues analysis suggests that data constitute a single cluster, positive samples are probably similar and, with high confidence, we can assume that they all belong to the target person. Otherwise, if the number of eigenvalues $\lambda_i > 0.9$ is greater than 1, data can be clustered in strongly different groups implying that an error has occurred in the Transductive Learning procedure and a negative result has been generated. In this case the last sample is rejected to avoid errors propagation.

5. Experimental Results

Fig. 2 depicts our complete proposal: the first step provides the use of a HOG based people detector [3] to obtain the bounding boxes containing people appearing in every video frame; subsequently Transductive Learning (Sec. 2) is used to find the target, comparing snapshots using the feature proposed in Sec. 3 and eventually updating the model when the target is found (Sec. 4). Our preliminary proposal is tailored for tracking a single target at a time and in our experiments tracker initialization is manually performed.

We evaluated our proposal on THIS and CAVIAR datasets, with a total number of approximately 40 video sequences. More precisely, experiments have been performed on *THIS project, Internal Videos, datasets 1 and 2* [1] and *CAVIAR Clips from a Shopping Center in Portugal, corridor view* [2]. Examined videos, depicted in Fig. 3, show people walking in public areas such as transport hubs or shopping centres. They represent challenging scenarios due to variations in pose and illumination of the target person, to the lack of variability of colors in dressing and to the difficulties to extract moving objects since most of the clips do not exhibit a clear background with no people in the scene.

In order to measure the quality of tracking results, we

[1] $http://imagelab.ing.unimore.it/visor$

[2] $http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1$

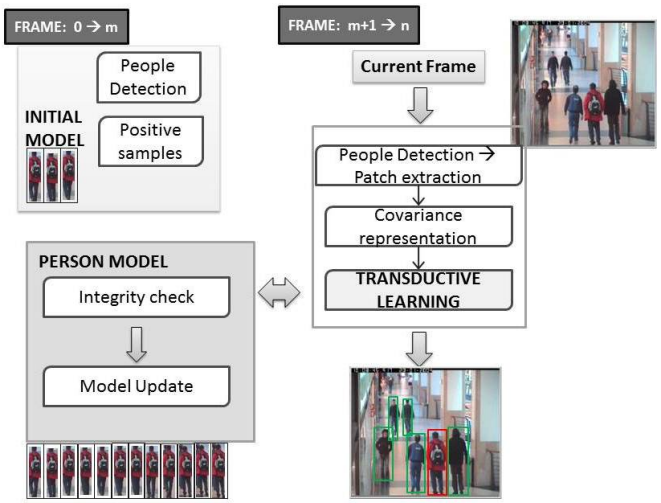

Figure 2. Overview of the whole tracking algorithm.

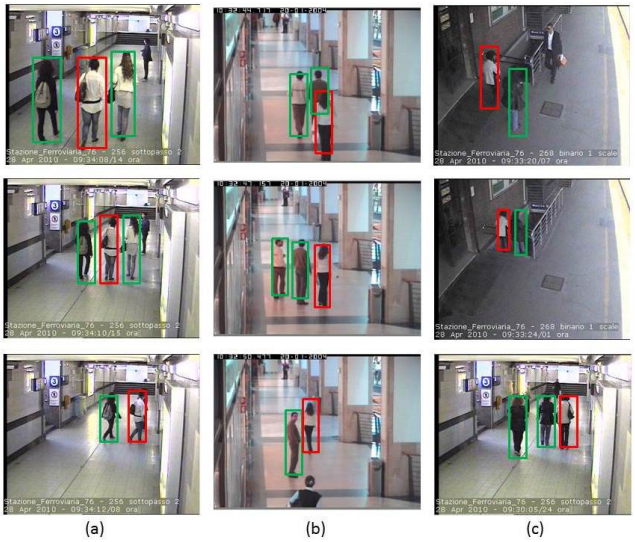

Figure 3. Examples of frames taken respectively from "THIS" and "CAVIAR" dataset. Rectangles show people found by the people detector and red rectangles underline the tracked person.

use the same metrics proposed in [6], according to which evaluated indexes are: Detection Rate (DR), Specificity (Spec), Accuracy (Acc), Positive Predicted Value (PPV), False Negative Rate (FNR), False Positive Rate (FPR) and Negative Predicted Value (NPV).

Tab. 1 contains the average indexes' values obtained respectively on THIS and CAVIAR datasets using our proposal and different state-of-the-art approaches. Referring to Tab. 1, the second and fifth columns offer the contrast with the Particle Filter based tracking in [4]. Our tracker exhibits higher values in accuracy, detection rate and specificity. This is mainly due to the transduction and the use of multiple images to build the target model that

	THIS			CAVIAR		
	PF	TLT	TLT NU	PF	TLT	TLT NU
DR	0.83	0.91	**0.92**	0.79	0.87	**0.90**
Spec	0.50	**0.99**	0.84	0.4	**0.98**	0.82
Acc	0.72	**0.89**	0.78	0.8	**0.92**	0.75
PPV	0.77	**0.95**	0.72	0.69	**0.90**	0.65
FNR	0.16	**0.09**	0.09	0.2	0.13	**0.09**
FPR	0.59	**0.01**	0.15	0.63	**0.01**	0.18
NPV	0.60	**0.88**	0.84	0.54	0.92	**0.99**

Table 1. THIS (T) and CAVIAR (C) results: Particle Filtering(PF), our Transductive Learning Tracker (TLT) and Transductive Learning Tracker without the Model Update(TLT NU) which is similar to the appearance tracker proposed in [20].

does not involve complex statistical functions. We would like to point out how in our experiments the initial model is composed of an average number of only three positive samples leading to good tracking performances. The fourth and seventh columns contain results obtained by our Transductive Learning Tracker without the integrity check, described in Sec. 4. Results underline how the model consistency check improves significantly the accuracy overcoming the problem of injecting wrong samples in the model. If an error is injected in the model the transducer tends to propagate the error during subsequent iterations, consequently the accuracy dramatically lowers. From a computational point of view the tracking algorithm runs, with the exception for the people detection phase, at 20 fps on an Intel Core i5 machine, using an average number of 30 input snapshots and 15 images for every target model, when five multiple instances of the tracker are used for tracking simultaneously five targets.

Finally we want to emphasize the test case given in the last column of Fig. 3. which represents an example of people reacquisition, where two different video sequences with the same persons are combined to test the effective capacity of the system in recognizing the same person after the occurrence of an occlusion or with different backgrounds.

6. Conclusions

In conclusion the system represents a new way for interpreting tracking as a snapshot searching problem using Transductive Learning to explore and learn about the target model during iterations. Performances on public surveillance datasets are encouraging and we aim at improving both, the accuracy and time performance of the algorithm by exploring the chance of modelling multiple classes during transduction, using motion information to strengthen the target assignment and adopting an iterative scheme to update the graph Laplacian in time without recomputing it for every frame.

References

[1] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *In Intl. Conf. on Machine Learning*, 2001.

[2] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller Meier, and L.J. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *Intl. Conf. on Computer Vision*, pages 1515–1522, 2009.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. of CVPR*, pages 886–893, 2005.

[4] A. Doucet and A.M. Johansen. A tutorial on particle filtering and smoothing: fifteen years later. In *In Handbook of Nonlinear Filtering*. University Press, 2009.

[5] O. Duchenne, J.Y. Audibert, R. Keriven, J. Ponce, and F. Segonne. Segmentation by transduction. In *Proc. of CVPR*, pages 1–8, 2008.

[6] T. J. Ellis. Performance metrics and methods for tracking in surveillance. In *PETS*, 2002.

[7] W. Forstner, B. Boudewijn Moonen, and C.F. Gauss. A metric for covariance matrices, 1999.

[8] T. Joachims. Transductive learning via spectral graph partitioning. In *Intl. Conf. on Machine Learning*, pages 290–297, 2003.

[9] M. Li, W. Chen, K. Huang, and T. Tan. Visual tracking via incremental self-tuning particle filtering on the affine group. In *Proc. of CVPR*, pages 1315–1322, 2010.

[10] Ulrike Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 2007.

[11] M.J. Metternich and M. Worring. Semi-interactive tracing of persons in real-life surveillance data. In *Proc. of MIFOR*, pages 43–48, 2010.

[12] F. Porikli and P. Pan. Regressed importance sampling on manifolds for efficient object tracking. In *Proc. of AVSS*, pages 406–411, 2009.

[13] O. Tuzel, F. Porikli and P. Meer. Region Covariance: A Fast Descriptor for Detection And Classification. In *Proc. of 9th ECCV*, pages 589–600, 2006.

[14] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[15] U. von Luxburg, M. Belkin, and O. Bousquet. Consistency of spectral clustering. *Annals of Statistics*, 2007.

[16] G. Wang, Y. Liu, and H. Shi. Covariance tracking via geometric particle filtering. *Intl. Conf. on Intelligent Computation Technology and Automation*, 1:250–254, 2009.

[17] Y. Wu, J. Cheng, J. Wang, and H. Lu. Real-time visual tracking via incremental covariance tensor learning. In *Intl. Conf. on Computer Vision*, pages 1631–1638, 2009.

[18] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38, 2006.

[19] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems*, pages 1601–1608. MIT Press, 2004.

[20] Y. Zha, Y. Yang, and D. Bi. Graph-based transductive learning for robust visual tracking. *Pattern Recogn.*, 43:187–196, 2010.