IDIAP RESEARCH REPORT


# EXPLOITING LONG-TERM CONNECTIVITY AND VISUAL MOTION IN CRF-BASED MULTI-PERSON TRACKING

Alexandre Heili  Adolfo Lopez-Mendez
Jean-Marc Odobez

Idiap-RR-05-2014

APRIL 2014

**Centre du Parc**, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11  F +41 27 721 77 12  info@idiap.ch  www.idiap.ch

# Exploiting Long-Term Connectivity and Visual Motion in CRF-based Multi-Person Tracking

Alexandre Heili*, *Student Member, IEEE,* Adolfo López-Méndez, and Jean-Marc Odobez, *Member, IEEE*

***Abstract*—We present a Conditional Random Field (CRF) approach to tracking-by-detection in which we model pairwise factors linking pairs of detections and their hidden labels, as well as higher order potentials defined in terms of label costs. To the contrary of previous works, our method considers long-term connectivity between pairs of detections and models similarities as well as dissimilarities between them, based on position, color and as novelty, visual motion cues. We introduce a set of feature-specific confidence scores which aim at weighting feature contributions according to their reliability. Pairwise potential parameters are then learned in an unsupervised way from detections or from tracklets. Label costs are defined so as to penalize the complexity of the labeling, based on prior knowledge about the scene like the location of entry/exit zones. Experiments on PETS'09, TUD, CAVIAR, Parking Lot and Town Centre public datasets show the validity of our approach, and similar or better performance than recent state-of-the-art algorithms.**

**EDICS Category: ARS-IVA**

## I. Introduction

Automated tracking of multiple people in video is a central problem in computer vision. It is particularly interesting in video surveillance contexts, where tracking the position of people over time might benefit tasks such as group and social behavior analysis, pose estimation or abnormality detection, to name a few. Nonetheless, multi-person tracking remains a challenging task, especially in single camera settings, notably due to sensor noise, changing backgrounds, high crowding, occlusions, clutter and appearance similarity between individuals.

Tracking-by-detection methods have become increasingly popular [19], [8], [36]. These methods aim at automatically associating human detections across frames, such that each set of associated detections univocally belongs to one individual in the scene. Compared to background modeling-based approaches, tracking-by-detection is more robust to changing backgrounds and moving cameras. However, human detection is not without weaknesses: detectors usually produce false alarms and they missdetect objects. Hence, on top of the numerous challenges of multiple person tracking, tracking-by-detection approaches must deal with detectors' caveats.

Several existing approaches address these issues by initially linking detections with high confidence to build track fragments or tracklets [19], [34], and then finding an optimal association of such tracklets. Although obtaining impressive results on several datasets, these approaches ultimately rely on low-level associations that are limited to neighboring time instants and reduced sets of features (color and adjacency). Hence, a number of higher-level refinements with different sets of features and tracklet representations are required in order to associate tracklets into longer trajectories.

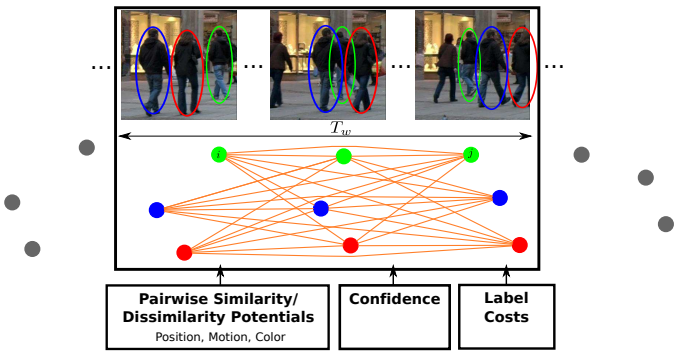


Fig. 1. Overview of the proposed approach. Detections in incoming frames are represented as observation nodes. Pairs of labels/observations within a temporal window $T_w$ are linked to form the labeling graph, thus exploiting longer-term connectivities (note: for clarity, only links having their two nodes within the shown temporal window are displayed). Pairwise feature similarity/dissimilarity potentials, confidence scores and label costs are used to build the energy function to optimize for solving the labeling problem within the proposed CRF framework.

In this paper, we explore an alternative approach that relies on longer-term connectivities between pairs of detections for multi-person tracking. We formulate tracking as a labeling problem in a Conditional Random Field (CRF) framework, where we target the minimization of an energy function defined upon pairs of detections and labels. Our approach is summarized in Fig. 1.

Contrarily to existing approaches, the pairwise links between detections are not limited to pairs of detections in adjacent frames, but between frames within a time interval $T_w$ (from $\pm 0.5$s to $\pm 2$s). Hence, the notion of tracklets is not explicitly needed to compute features for tracking, allowing us to keep the optimization at the detection level. In particular, a novelty of our approach is to directly use the visual motion computed from the video sequence for data association. This avoids resorting to tracklet creation or cumbersome tracklet hypothesizing and testing optimization to obtain discriminative motion information.

Another differential trait of our method is the form of energy potentials, formulated here in terms of similarity and dissimilarity between pairs of detections. Moreover, the proposed potentials depend not only on sets of features, but also on the time interval between two detections. In this way, we model how discriminative a feature is given the observed

* Corresponding author

The authors are with Idiap Research Institute, Rue Marconi 19, 1920 Martigny, Switzerland. A. Heili and J-M. Odobez are with École Polytechnique Fédérale de Lausanne, Switzerland. *E-mail*: {aheili, adolfo.lopez-mendez, odobez}@idiap.ch

distance in the feature space and the time gap between pairs of detections, an important characteristic when considering long-term connectivity. Furthermore, to take into account not only the actual feature distance value but also its reliability, we exploit a set of confidence scores per feature to characterize how trustable the pairwise distances are. For instance, visual cue distances are given a lower confidence whenever one of the detections is possibly occluded. These scores ultimately allow to re-weight the contribution of each feature based on spatio-temporal cues, and to rely on the most reliable pairwise links for labeling. This is important near occlusion situations, where thanks to long-term connectivity, the labeling can count on cleaner detections just before or after occlusion to propagate labels directly to the noisier detections obtained during occlusion instead of through adjacent drift-prone frame-to-frame pairwise links only.

One important advantage of our modeling scheme is that it allows to directly learn the pairwise potential parameters from the data in an unsupervised and incremental fashion. To that end, we propose a criterion to first collect relevant detection pairs to measure their similarity/dissimilarity statistics and learn model parameters that are sensitive to the time interval between detection pairs. Then, at a successive optimization round, we can leverage on intermediate track information to gather more reliable statistics and exploit them to estimate accurate model parameters.

Finally, compared to some existing CRF approaches for tracking [34], [36], [17] a novel aspect of our framework is that the energy function includes higher order terms in the form of label costs. The aim of such label costs is to model priors on label fields. In our tracking framework, this translates into penalizing the complexity of the labeling, mostly based on the fact that sufficiently long tracks should start and end in specific areas of the scenario. We are interested in static camera settings, in which scene-specific maps can be defined for that purpose.

To summarize, the paper addresses the multi-person tracking problem within a tracking-by-detection approach and makes contributions in the following directions (see also Fig. 1):

1) A CRF framework formulated in terms of similarity/dissimilarity pairwise factors between detections and additional higher-order potentials defined in terms of label costs. Differently from existing CRF frameworks, our method considers long-term connectivity between pairs of detections. Note however that long-term temporal connectivity alone is generally not sufficient to guarantee good results, and needs to be exploited in conjunction with the other contributions described below: visual motion, confidence weights, time-sensitive parameters with unsupervised learning from tracklets.
2) A novel potential based on visual motion features. Visual motion allows incorporating motion cues at the bottom association level, i.e., the detection level, rather than through tracklet hypothesizing.
3) A set of confidence scores for each feature-based potential and pair of detections. The proposed confidence scores model the reliability of the feature considering spatio-temporal reasoning such as occlusions between detections.
4) Thanks to the similarity/dissimilarity formulation, the parameters defining the pairwise factors can be learned in an unsupervised fashion from detections or from tracklets, leading to accurate time-interval dependent factor terms.

Experiments conducted on standard public datasets show the benefit of the different modeling contributions. They demonstrate that our optimization conducted at the detection node level but relying on longer time window association leads to competitive performance compared to recent state-of-the art methods.

The paper is structured as follows. Section II describes related work. The CRF framework is formulated in Section III. Pairwise potentials with associated confidence scores are detailed in Section IV whereas label costs are described in Section V. Unsupervised parameter learning is explained in Section VI. Section VII describes the optimization methodology. Finally, experimental results are presented in Section VIII.

## II. Related Work

Tracking-by-detection methods have become increasingly popular in the vision community. On the contrary to generative methods, detection-based trackers use a discriminative classifier to assess the presence of an object in a scene, which is generally more robust, as state-of-the-art detectors give very good performance at detecting humans [13][15]. The detector's output is used to generate target hypotheses in each frame, which then have to be transitively linked to form trajectories with consistent identity labels. Tracking-by-detection can therefore be formulated as a data association problem. This data association generally relies on affinity models between detections in successive frames based on motion constraints and intrinsic object descriptors such as color [38].

The association problem is addressed by some approaches on a multi-frame basis [27], [33], [3]. Dependencies are often modeled using graphs, and the optimization problem then consists in finding the best paths between all the detections in separate frames. The process can be applied on potentially large time windows, so as to overcome the sparsity in the detection sets induced by missed detections and also to deal with false alarms, but the complexity of the optimization increases rapidly. Moreover, due to the temporal locality considered in this context, tracking-by-detection techniques can perform poorly in presence of long-term occlusions, i.e. many successive missed detections.

Alternatively, to reduce the computational cost and to progressively increase the temporal range for correspondences, hierarchical approaches can be considered, in which low-level tracklets are first generated and then merged at a higher-level. For instance, in [19], the lower level associates pairs of detections in adjacent frames based on their similarity in position, size and appearance. The resulting tracklets are then fed into a Maximum A Posteriori (MAP) association problem which is solved by the Hungarian algorithm, and further refined at a higher level to model scene exits and

occluders. As there are fewer tracklets than detections, the complexity of the optimization is reduced, but any wrong association made at the low-level is then propagated to the next hierarchy level. This hierarchical association is also followed in the CRF models presented in [34], [36]. The motivation of the CRF framework is to introduce pairwise potentials between tracklets, such that pairs of difficult tracklets can be better distinguished. While [22], [35] make emphasis on learning discriminative appearance models for tracklets, they both follow the hierarchical association of [19]. Similarly, Bak et al. [5] propose a two-level association algorithm where tracklets are linked by using discriminative analysis on a Riemannian manifold. The described methods rely on bottom level associations between pairs of detections in consecutive frames, often with a subset of features (motion information is not used at the bottom level). This limitation can be critical, provided that errors are propagated to higher levels of the hierarchy.

A different approach to hierarchical association of detections is presented in [39]. To generate the first level tracklets, detections within predefined short time windows are linked, thus breaking the frame adjacency constraint of previously described methods. Then, tracklet association between consecutive windows is performed. At both levels, the same optimization framework is employed. The objective function relies on a motion model where all pairs of detections within the tracklet contribute to build a motion estimate which can be used with a constant speed assumption to compute a prediction error. Additionally, a virtual detection generation approach is proposed in order to tackle occlusions.

Alternatively, some authors focus on global methods that aim at alleviating these short temporality limitations. They usually consider the whole span of the sequence, which can be a problem if online processing is required. In [40], the authors use a similar MAP formulation as in [19] but embed it in a network framework where min-cost flow algorithm can be applied. The authors of [8] formulate the problem as finding the flow of humans on a discrete grid space that minimizes the cost of going through the detections, which are obtained by fusing the foreground information from several camera views. In [29], the authors extend their method by adding global appearance constraints. Impressive results are obtained, but only results in indoor scenarios are shown, where relatively clean detections from multiview background subtraction images are used. Furthermore, in many tracking scenarios, multiple synchronized and calibrated cameras are not available.

Labeling detections with identity labels can also be done jointly with finding smooth trajectories that best explain the data. The method proposed in [4] tackles the problem by alternating between discrete data association and continuous trajectory estimation using global costs. This method relies solely on trajectories and does not involve appearance of objects.

Some multi-person tracking algorithms focus on context learning and model adaptation in order to address possible limitations of pre-learned affinity models. Context models proposed in [37][31] rely on the availability of sufficient training data. If such data cannot be acquired, one can alternatively adapt tracking models by using local crowd density estimations [28]. Similarly, [30] propose a tracklet adaptation approach based on the variance of the observed features along a path. In [24], contextual cues such as target births and clutter intensities are incrementally learned using tracker feedback.

Different from the above, we benefit from important temporal context by connecting detection pairs not only between adjacent frames, but between frames within a long time interval. Not only we differentiate from [39] in that we exploit longer-term connectivities between detections, but also in that our method is built entirely on pairwise links between detections, allowing us to re-label detections at any iteration of the algorithm. Since the notion of tracklet is not explicitly used in the proposed framework, we use motion information by introducing a novel feature based on visual motion. Furthermore, to the contrary of most existing methods above, our approach does not only optimize the label field on a similarity hypothesis basis, but also relies on a dissimilarity information to assess the labeling. By contrasting the two hypotheses for each detection pair, the model is more robust to assess the appropriateness of a given association. Apart from the larger connectivity between pairs of detections, our CRF framework differs from [34], [36] in that we consider confidence scores for the features, as well as higher order potentials in the form of label costs. Confidence scores can be regarded as a context adaptation approach where, differently from methods such as [30], we do not rely on tracklets but on the position of detections on a per-frame basis.

## III. CRF Tracking Framework

This Section introduces the main elements of our tracking framework. We start by introducing our data representation, and then present how we formulate our tracking problem. A list of all symbols used in the manuscript, along with their brief definition and where they are introduced in the paper is given in the supplementary material.

### A. Data representation

Let us define the set of detections of a video sequence as $R = \{r_i\}_{i=1:N_r}$, where $N_r$ is the total number of detections. The features we choose to represent our detections are articulated around 3 cues: position, motion and color. More precisely, each detection is defined as

$$r_i = (t_i, \mathbf{x}_i, \mathbf{v}_i, \{\mathbf{h}_i^b\}^{b\in\mathcal{P}}) \tag{1}$$

which comprises the following features:

- $t_i$ denotes the time instant at which the detection occurs;
- $\mathbf{x}_i$ denotes the 2D image or ground-plane position depending on the availability of calibration information;
- $\mathbf{v}_i$ denotes the 2D image plane visual motion computed from the video sequence;
- $\mathbf{h}_i^b$ with $b \in \mathcal{P} = \{\text{whole}, \text{head}, \text{torso}, \text{legs}\}$ denotes a set of multi-resolution color histograms extracted from a set $\mathcal{P}$ of body parts.

Note that, in contrast to existing approaches, each detection has an associated motion vector $\mathbf{v}_i$, which is independent

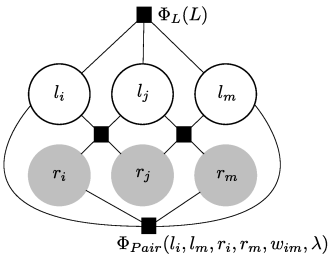


Fig. 2. Factor graph illustration of our Conditional Random Field model.

of the label field, i.e., motion is not derived from tracklets but from detections. In our case, a robust estimation of this motion is conducted by performing a weighted average of the displacement estimated at several body part patches resulting from a part-based human detector, where the weight of each displacement vector indicates the motion reliability based on the matching distance and how uniform the patch is. For the color descriptors, we define parts that represent the whole detection region as well as three different spatial regions (head, torso, legs) to take advantage of both a holistic representation and heuristically defined body parts. Further implementation details are given in Section VIII-B.

### B. Problem formulation

We formulate multi-object tracking as a detection labeling problem, in which we seek for the optimal label field $L = \{l_i\}_{i=1:N_r}$, where $l_i$ denotes the label of detection $r_i$, so that detections within a same track should be assigned the same label. Labels can take their values in $\mathbb{N}$ as we do not know in advance the number of objects in the scene.

To solve this labeling task, we rely on a CRF formulation. Assuming the graphical model and factor graph shown in Fig. 2, we model the posterior probability of the label field given all the observations as follows:

$$p(L|R,\lambda) = \frac{1}{Z(R)}\Phi_{Pair}(L,R,\mathcal{W},\lambda)\Phi_L(L) \quad (2)$$

$$\propto \left(\prod_{(i,j)\in\mathcal{I}}\prod_{k=1}^{N_f}\Phi_k(l_i,l_j,r_i,r_j,w_{ij}^k,\lambda^k)\right)\Phi_L(L) \quad (3)$$

where $\mathcal{I}$ denotes the set of connected detection pairs, for each detection pair we introduce $N_f$ factor terms $\Phi_k$ to account for different pairwise feature similarity/dissimilarity measurements, $\lambda = \{\lambda^k\}$ denotes the set of parameters associated with each of these factors, and $\mathcal{W} = \{w_{ij}^k\}$ with $w_{ij}^k \in [0,1]$ denotes the set of confidence scores associated with each feature and detection pairs. In contrast to [18] that only considered pairwise terms, the above formulation incorporates a prior $\Phi_L$ over label fields in terms of higher-order potentials. This prior acts as regularizers penalizing complex solutions, and will be detailed in Section V.

### C. Factor modeling.

The factors $\Phi_k$ are modeled using a *long-term*, *two-hypothesis*, *time-interval dependent* and *confident* pairwise approach, as explained below. Firstly, we limit the number of detection pairs $(r_i, r_j)$ to be considered by imposing a *long-term* connectivity constraint:

$$\mathcal{I} = \{(i,j) \ / \ 1 \leq \mathbf{\Delta}_{ij} = |t_j - t_i| \leq T_w\}. \quad (4)$$

where $T_w$ is our long term window size. Secondly, for each factor term, a feature function $f_k(r_i, r_j)$ is defined that computes a distance measure between detection characteristics. Then, the corresponding CRF pairwise factor is defined as:

$$\Phi_k(l_i,l_j,r_i,r_j,w_{ij}^k,\lambda^k) \triangleq p(f_k(r_i,r_j)|H(l_i,l_j),\lambda^k_{\mathbf{\Delta}_{ij}})^{w_{ij}^k}. \quad (5)$$

where the symbol $\triangleq$ means by definition. This factor depends on the distribution $p(f_k|H,\lambda^k_{\mathbf{\Delta}})$ of the feature distance $f_k$ under *two different hypotheses* corresponding to whether the labels are the same or not, that is:

$$H(l_i,l_j) = \begin{cases} H_0 & \text{if } l_i \neq l_j \\ H_1 & \text{if } l_i = l_j \end{cases} \quad (6)$$

Furthermore, the feature distribution under the two hypotheses is *time-interval sensitive*, in the sense that we define such a distribution for each time interval $\mathbf{\Delta}$ that can separate two detections. This allows to take into account the evolution of the feature according to this time parameter. In the model, the dependency is introduced thanks to the use of different sets of parameters $\lambda^k_{\mathbf{\Delta}}$ for each interval $\mathbf{\Delta}$.

Finally, the factor $\Phi_k$ defined by Eq. 5 accounts for the *confidence* $w_{ij}^k$ we have between detection pairs by powering the feature distribution with $w_{ij}^k$. Intuitively, lower confidence values will flatten the distribution of a feature leading to less discriminative potential, lowering the factor difference under the two hypotheses. At the limit, if $w_{ij}^k = 0$, the factor of a given feature distance will be identical (equal to one) under the two hypotheses.

### D. Equivalent Energy minimization.

Our goal is to optimize the probability defined by Eq. 3. Given our factor definition (Eq. 5) and since the confidence scores are independent of the hypothesis $H(l_i,l_j)$, we can divide the expression of Eq. 3 by Cst $= \prod_{(i,j)\in\mathcal{I}}\prod_k p(f_k(r_i,r_j)|H_0,\lambda^k_{\mathbf{\Delta}_{ij}})^{w_{ij}^k}$. By further taking the negative logarithm of the resulting expression, the maximization of Eq. 3 can be equivalently conducted by minimizing the following energy:

$$U(L) = \left(\sum_{(i,j)}\sum_{k=1}^{N_f} w_{ij}^k\,\beta_{ij}^k\,\delta(l_i - l_j)\right) + \Lambda(L) \quad (7)$$

where $\delta(.)$ denotes the Kronecker function ($\delta(a) = 1$ if $a = 0$, $\delta(a) = 0$ otherwise), the Potts coefficients for each pairwise link and each feature distance are defined as:

$$\beta_{ij}^k = \log\left[\frac{p(f_k(r_i,r_j)|H_0,\lambda^k_{\mathbf{\Delta}_{ij}})}{p(f_k(r_i,r_j)|H_1,\lambda^k_{\mathbf{\Delta}_{ij}})}\right], \quad (8)$$

and the term $\Lambda(L) = -\log(\Phi(L)$ represents the label cost.

As can be seen, for each feature, the Potts coefficients are defined by the loglikelihood ratio of the feature distance of a detection pair under the two hypotheses. Since in the energy

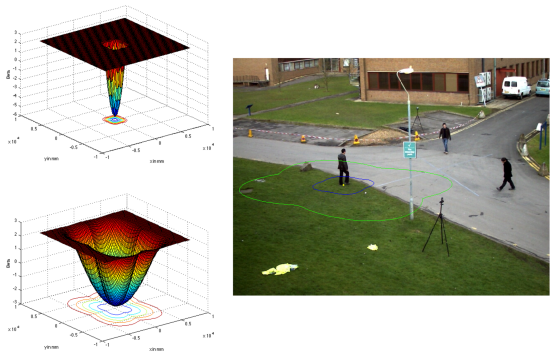

Fig. 3. Left: The $\beta$ surface and iso-contours (below) for the position model for $\mathbf{\Delta} = 3$ (top) and $\mathbf{\Delta} = 15$ (bottom). Right: the iso-contour of value 0 of the $\beta$ surface for $\mathbf{\Delta} = 3$ (blue) and $\mathbf{\Delta} = 15$ (green), centered around one detection.

of Eq. 7, the terms for pairs having different labels ($l_i \neq l_j$) vanish and only those for which $l_i = l_j$ remain, the Potts coefficient can be seen as "costs" for associating a detection pair within the same track. When $\beta_{ij}^k < 0$, the more *negative* this coefficient will be, the more likely the pair of detections *should* be associated, so as to minimize the energy in Eq. 7. Reversely, when $\beta_{ij}^k > 0$, the more *positive* this coefficient will be, the more likely the pair of detections *should not* be associated, so as to minimize the energy in Eq. 7. When $\beta_{ij}^k = 0$, there is no preference for associating or not the pairs.

In the following Section, the specific features, factor models and confidence scores will be defined and illustrated. The label cost term $\Lambda(L)$ will be defined in Section V.

## IV. Similarity/Dissimilarity Confident Factor Modeling

The previous Section introduced our general modeling approach. In this Section, we specify more precisely the different pairwise feature functions $f_k$ that we have considered along with their associated distributions and the parameters that characterize them. In practice, we used $N_f = 7$ feature functions constructed around three cues: position, motion and color. Their definitions are provided in Subsections IV-A, IV-B and IV-C, while IV-D summarizes the model parameters that specify them and that will be learned automatically. In a second stage (Subsection IV-E) we will present the pairwise confidence scores $w_{i,j}^k$ that are used to weight the contribution of each factor term of a detection pair in the overall energy. Note that the focus of this Section is on the design of the similarity distributions, and that parameter learning will be described later in Section VI.

### A. Position cue similarity distributions

The position feature is defined for $k = 1$ as $f_1(r_i, r_j) = \mathbf{x}_i - \mathbf{x}_j$. We assume that its probability follows a Gaussian distribution with 0 mean and whose covariance depends on the two label hypotheses $H_0$ or $H_1$ and also *on the time gap* $\mathbf{\Delta}_{ij} = |t_i - t_j|$ between the detection pairs:

$$p(f_1(r_i,r_j) = f | H(l_i,l_j) = H, \lambda^1) = \mathcal{N}(f\ ;\ 0, \Sigma^H_{\mathbf{\Delta}_{ij}}) \quad (9)$$


Fig. 4. Role of the visual motion for tracking. Left: Detection $r_i$ at time $t_i$ along with its estimated visual motion $\mathbf{v}_i$ (green ellipse). Right: in subsequent frames, the motion cost associated to this detection favors associations with other detections located in the direction of motion (shaded area) and penalizes associations in opposite directions (example of blue person, gray ellipse).

Fig. 3 illustrates for two different time intervals the learned models in the form of the Potts coefficient $\beta$ in function of the distance $(dx, dy)$ between detection pairs. As expected, $\beta$ is highly negative for distance features close to 0 and increases with the distance. The iso-contours of the $\beta$ surface are also shown. Amongst them, the zero-contour is a good indicator of the learned model, as it shows the frontier between the domain where hypothesis $H_1$ prevails and the one where $H_0$ prevails. Fig. 3 displays them centered around one detection $r_0$ for two different values of $\mathbf{\Delta}$. After $\mathbf{\Delta} = 3$ frames, any detection that falls within the blue contour will vote strongly for the association with $r_0$ (negative cost). After $\mathbf{\Delta} = 15$ frames (around 2 seconds in this case), the model is more relaxed and favors association within the green contour.

### B. Visual motion cue similarity distributions

The position distance similarity alone does not exploit any directional information and can lead to ambiguities. In order to use an estimation of movement direction at the detection level, we propose to exploit visual motion. In our formalism, the visual motion information is represented by two feature functions $f_k$ with $k \in \{2, 3\}$ defined as follows:

$$f_2(r_i,r_j) = \frac{\mathbf{v}_i.\mathbf{d}_{ij}}{\|\mathbf{v}_i\|\|\mathbf{d}_{ij}\|} \text{ and } f_3(r_i,r_j) = \frac{\mathbf{v}_j.\mathbf{d}_{ij}}{\|\mathbf{v}_j\|\|\mathbf{d}_{ij}\|}. \quad (10)$$

where $\mathbf{d}_{ij} = \mathbf{x}_j^{im} - \mathbf{x}_i^{im}$ denotes the displacement between the image positions of the detections[1] and $\mathbf{v}_i$ or $\mathbf{v}_j$ correspond to their respective visual motion. Given a pair of detections (with $t_j > t_i$), they represent the cosine between their image displacement (as measured by $\mathbf{d}_{ij}$) and the visual motion $\mathbf{v}_i$ or $\mathbf{v}_j$. Intuitively, for detections belonging to the same track, these vectors should be aligned (with a cosine close to 1). The computation of the visual motion vectors will be presented in Section VIII-B.

For the motion feature distribution, we discretized the cosine and used a non-parametric approach by assuming for each time gap $\mathbf{\Delta}$ and hypothesis $H$ that the features follow a multinomial distribution of parameters $\alpha_{\mathbf{\Delta},H}$:

$$p(f_k(r_i,r_j) = f | H(l_i,l_j) = H, \lambda^k) = \alpha_{|t_i - t_j|,H}(b(f)) \quad (11)$$

where $b(f)$ denotes the bin index associated with the cosine after quantization. Since $f_2$ and $f_3$ play exactly the same role, we used the same model and parameters for both of them.

The intuition is illustrated in Fig. 4: detections with the same labels are unlikely to fall outside a 2D cone spanned

[1]Note that $\mathbf{x}^{im}$ corresponds to $\mathbf{x}$ when no calibration is available, see Subsec. III-A.

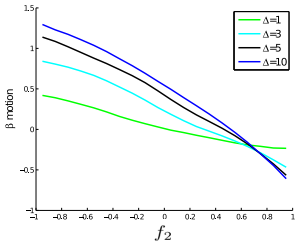


Fig. 5. Motion feature: learned $\beta$ curves on CAVIAR for different time intervals.

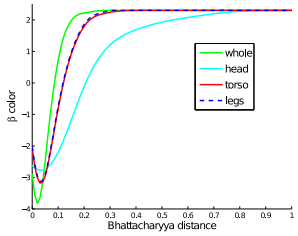


Fig. 6. Learned $\beta$ curves on PETS for different body parts subject to a time interval of 15 frames.

by observed motion vectors. This is confirmed by the beta curves automatically learned from data shown in Fig. 5, which favor association when motion and detection displacements are aligned (cosine near 1) and becomes more positive as the cosine becomes lower than $\approx 0.5$, discouraging association. Interestingly, we see that the model is more discriminative for larger time gaps $\mathbf{\Delta}$, when the uncertainty about the displacement (measured from the detected position) is lower.

### C. Color cue similarity distributions

Finally, we propose an appearance similarity measure based on Bhattacharyya distances $D_h$ between color histograms. The pairwise color features are defined for $k \in [4, 7]$ as:

$$f_k(r_i, r_j) = D_h(\mathbf{h}_i^{g(k)}, \mathbf{h}_j^{g(k)}) \tag{12}$$

where $g$ is a mapping between color feature indices and corresponding body parts: $g : k \in [4, 5, 6, 7] \rightarrow g(k) \in [\text{whole}, \text{head}, \text{torso}, \text{legs}]$. Then, the distribution of each feature $f_k$ for a given hypothesis $H$ and time gap $\mathbf{\Delta}$ is assumed to follow a Gaussian mixture model (GMM) given by:

$$p(f_k(r_i, r_j) = f | H(l_i, l_j) = H, \lambda^k) = \sum_{n=1}^{N_{mix}} \pi^{H,k}_{\mathbf{\Delta}_{ij},n} \mathcal{N}(f | \mu^{H,k}_{\mathbf{\Delta}_{ij},n}, \sigma^{H,k}_{\mathbf{\Delta}_{ij},n}) \tag{13}$$

with $\mathbf{\Delta}_{ij} = |t_j - t_i|$ and $N_{mix} = 10$ represents the number of mixture components. In practice, GMM parameters $\lambda^{H,k}_{\mathbf{\Delta}} = \{\pi^{H,k}_{\mathbf{\Delta},n}, \mu^{H,k}_{\mathbf{\Delta},n}, \sigma^{H,k}_{\mathbf{\Delta},n}, n \in [1, \ldots, N_{mix}]\}$, i.e. weights, means and variances, are estimated using Expectation-Maximization from appropriate training data (cf. the unsupervised parameter learning Section VI).

Fig. 6 illustrates the resulting learned $\beta$ models for different body parts under a time interval $\mathbf{\Delta}$ of 15 frames. It can be seen that for small Bhattacharyya distances between detection pairs, the association cost is negative and progressively rises as the distance increases, reaching positive values where it disfavors association. Surprisingly, the torso and legs regions exhibit almost no difference in their learned $\beta$ curves. The head region shows less discrimination, which might be understandable since at the considered resolution, the head of people contains few distinctive color feature. Note that color models also exhibit time-interval dependencies, as illustrated in Fig. 9, where the $\beta$ curves of the torso part are displayed for two different values of the time gap $\mathbf{\Delta}$.

### D. Similarity/dissimilarity distribution parameters

The parameters $\lambda = \{\lambda^k\}$ of the similarity and dissimilarity functions are thus defined for each feature $k$ as $\lambda^k = \{\lambda^k_{\mathbf{\Delta}}, \mathbf{\Delta} = 1 \ldots T_w\}$, with: $\lambda^1_{\mathbf{\Delta}} = \{\Sigma^{H_0}_{\mathbf{\Delta}}, \Sigma^{H_1}_{\mathbf{\Delta}}\}$ for the position feature (k=1), $\lambda^k_{\mathbf{\Delta}} = \{\alpha_{\mathbf{\Delta},H_0}, \alpha_{\mathbf{\Delta},H_1}\}$ for the motion feature (k=2,3), and with $\lambda^k_{\mathbf{\Delta}} = \{\lambda^{H_0,k}_{\mathbf{\Delta}}, \lambda^{H_1,k}_{\mathbf{\Delta}}\}$ for the color feature (k=4,5,6,7). It is worth emphasizing that each factor is time-interval sensitive, as the parameters depend on the time between the detection pairs.

### E. Pairwise factor contextual weighting

The energy terms defined previously rely on feature distance distributions whose parameters are learned in an unsupervised way as explained in Section VI. These distributions, however, are global and only reflect the overall feature distance statistics and their discriminative power. To leverage on the local context during test time, we have introduced the weights $w^k_{ij}$ in the definition of our factor terms and of the resulting energy function (7). For each feature $k$ and detection pair $r_i$ and $r_j$, they allow to modulate the previously defined energy terms according to the knowledge of the detection's spatial surroundings.

For instance, when some detection bounding boxes overlap within a frame, the collected color measurements might be corrupted. Hence, we should strongly downvote the color feature contribution of the occluded detections according to the importance of the coverage. Similarly, the visual motion is measured from pixel displacements and such detection overlaps can lead to inaccurate motion estimates that we do not want to rely on for association. By downweighting the contribution of the color and motion features in such cases, we avoid taking into account unreliable features, but can still rely on more accurate measurements done before or after the occlusion and on the position feature to track a partially occluded object. Following the above intuition, the weights have been defined as described below.

**Color factor weighting.** Let us define the confidence $c(r_i)$ of the visual cues of a detection $r_i$ based on the overlap with the other detections occurring in the same frame $t_i$ as:

$$c(r_i) = 1 - \min\left(1, \sum_{\substack{r_j \neq r_i \\ t_j = t_i}} \frac{A(r_i \cap r_j)}{A(r_i)}\right) \tag{14}$$

where $A(r)$ denotes the area defined by the region associated with the detection $r$. As can be seen, this confidence is

maximum (equal to 1) when the detection does not overlap with any other detection, and decreases in function of the degree of overlap. Accordingly, for each of the color cue ($k = 4, 5, 6, 7$), we simply define the pairwise confidence score as the geometric average of the individual detection confidences, divided by 4 (the number of features for the color cue) to have a normalized confidence score per cue:

$$w_{ij}^k = \frac{\sqrt{c(r_i)c(r_j)}}{4}, \forall k \in \{4, 5, 6, 7\}. \quad (15)$$

**Motion factor weighting.** We used a similar approach for this cue. However, since the reliability of an estimated motion $\mathbf{v}_i$ only depends on the region of the detection $r_i$ it is computed on, we have defined the confidence score for the motion feature implying $\mathbf{v}_i$ ($k = 2$) and $\mathbf{v}_j$ ($k = 3$) as follows:

$$w_{ij}^2 = \frac{c(r_i)}{2} \text{ and } w_{ij}^3 = \frac{c(r_j)}{2}. \quad (16)$$

**Position factor weighting.** Finally, we also introduced a confidence score aiming at downscaling the position energy term for large time intervals. Indeed, as the time difference $\mathbf{\Delta}$ between two detection increases, the reliability of the position similarity for associating them decreases. This is particularly true in crossing scenarios or when two persons follow each other: in both cases, one of the person's trajectory passes near the other person's previous locations $\mathbf{\Delta}$ time steps ago, and these small distances tend to vote in favor of association. In order to avoid this effect, we have reduced the contribution of the energy term for larger time intervals by defining the confidence score of the position model as:

$$w_{ij}^1 = \frac{1}{1 + e^{|t_i - t_j| - \theta_f}} \quad (17)$$

where $\theta_f$ denotes the time separation at which the confidence starts to decrease: below $\theta_f$, the confidence is near 1; at $\theta_f$ it is equal to 0.5, and beyond it tends to 0 as the time gap $|t_i - t_j|$ increases.

## V. Label Costs

The energy terms defined earlier concerned detection pairs and did not allow to reason at the global level. The label cost $\Lambda(L)$ we introduced in our energy function of Eq. 7 allows to do so by penalizing model complexity. That is, its goal is to avoid having too many labels and obtain coherent tracks from the scene viewpoint. Intuitively, this means that real tracks should start and end near scene entrance/exits (scene boundaries), and that therefore, tracks should be penalized for starting or ending within the scene. Note however, that this is true only for long-enough tracks: short ones, that are less reliable and that are likely to correspond to false alarms should not be penalized.

Before defining the label cost, let us introduce the following notations. For each unique label $l$, we can define its associated track $\tau_l = \{r_i \ / \ l_i = l\}$ along with its main characteristics: its start time $t_l^s = \min\{t_i \ / \ r_i \in \tau_l\}$, its end time $t_l^e = \max\{t_i \ / \ r_i \in \tau_l\}$, its duration $d_l = t_l^e - t_l^s$, and finally its start and end locations defined by $\mathbf{x}_{t_l^s} = \{\mathbf{x}_i \ / \ r_i \in \tau_l, t_i = t_l^s\}$ and $\mathbf{x}_{t_l^e} = \{\mathbf{x}_i \ / \ r_i \in \tau_l, t_i = t_l^e\}$, respectively.

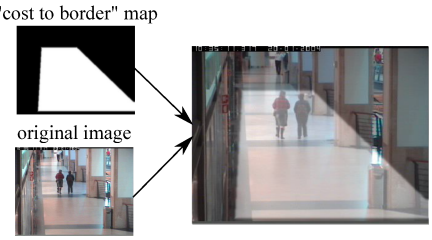


Fig. 7. Label cost illustration for the CAVIAR data. Long enough tracks starting or ending in the light regions will be penalized. See text for more details.

Then, to achieve the objectives qualitatively stated earlier, we have defined the label cost as follows:

$$\Lambda(L) = \rho \sum_{l \in \mathcal{U}(L)} (C^s(\tau_l) + C^e(\tau_l)) \quad (18)$$

where $\mathcal{U}(L)$ denotes the set of unique labels comprised in the label field $L$, the parameter $\rho$ controls the importance of the label cost with respect to the pairwise energies, and the start and ending costs of an individual track are defined as:

$$\begin{aligned} C^s(\tau_l) &= D(d_l)(\, B(\mathbf{x}_{t_l^s}) S(t_l^s - t_0; \theta_{tm}) \,) \\ C^e(\tau_l) &= D(d_l)(\, B(\mathbf{x}_{t_l^e}) S(t_{end} - t_l^e; \theta_{tm}) \,) \end{aligned} \quad (19)$$

where $\theta_{tm}$ is a temporal parameter related to the proximity to the start $t_0$ and end $t_{end}$ of the sequence, and the different terms of this expression that we explain below implement the intuition described earlier.

First of all, the function $B(\mathbf{x}) \in [0, 1]$ represents the cost of starting or ending a track at location $\mathbf{x}$, and is illustrated in Fig. 7. In practice, we define some scene border regions inside which starting or ending a track has no cost ($B(\mathbf{x}) = 0$) (dark region in Fig. 7). On the contrary, tracks that start or end far from these borders have a higher cost ($B(\mathbf{x}) = 1$) (light regions in Fig. 7). Smooth transitions between these regions are obtained through filtering. However, since people may already be in the scene at the beginning of the sequence, tracks that start far from the border at this moment should not be penalized. This is achieved thanks to the sigmoid term:

$$S(t_l^s - t_0; \theta_{tm}) = \frac{1}{1 + e^{-\left((t_l^s - t_0) - \theta_{tm}\right)}}$$

which is close to 0 for $t_l^s$ near $t_0$ and tends to 1 as $t_l^s$ increases. A similar treatment is done for tracks that end by the end of the sequence, since people might still be in the scene at that moment.

Finally, since short tracks that are less reliable might be due to false alarms, they should not be too much penalized to avoid encouraging their association. Thus the overall cost is modulated according to the track duration:

$$D(d_l) = \min(d_l, d_{\max}) \quad (20)$$

where $d_{\max}$ is a saturation value beyond which a track is considered long enough to be reliable, and all tracks are penalized in the same way.

## VI. Unsupervised Parameter Learning

The appropriate setting of the model parameters is of crucial importance for achieving good tracking results, but can be a tedious task. We remind that since distributions exhibited time dependencies, we have defined our models to be time-sensitive and feature-specific, which means that parameters need to be defined for each feature and each time interval up to $T_w$. Moreover, parameters also depend on the two-fold hypothesis $H$, so that ultimately, we have a large parameter space size. In practice, one would like to avoid supervised learning, as this would require tedious track labeling for each scene or camera.

In the following we propose an approach for learning the factor parameter set in an unsupervised fashion. More precisely, the first step is to learn model parameters by relying directly on the raw detections within training videos of a given scene. For convenience, we denote with a $\star$ superscript the notations that apply to these initial models (for instance, these models are learned up to $T_w^\star$). These models can be used for tracking on these training videos, and, provided we use a low $T_w^\star$ value, can lead to pure tracklets [17].

Thus, in a second step, these tracklets corresponding to an intermediate labelling $L^\star$ can be conveniently used to refine model parameters and learn parameters for larger $T_w$ values. The process could then be iterated (use new learned parameters for tracking, then resulting tracklet for parameters learning), but experiments showed that in general no further gain can be achieved.

In this paper, since we consider rather short sequences for testing, unsupervised learning is performed in batch mode directly on the test sequence, i.e. the training set is the whole test sequence, except for the CAVIAR dataset, in which we use as training videos the set of 6 videos that are not used in the test. The overall procedure of unsupervised batch learning and tracking is summarized in the block diagram of Fig. 8. More details are provided below.

**Unsupervised Learning from Detections.** Learning the model parameters $\lambda$ can be done in a fully unsupervised way using a sequence of detection outputs.

When no labels are provided, the intuition for learning consists of collecting training data as follows: for a given detection at time $t$, the closest detection amongst the detections at time $t + \mathbf{\Delta}$ should statistically correspond to a detection of the same track, while the second closest detection[2] would correspond to a different person. Thus, for each time gap $\mathbf{\Delta}$, we collect for each detection its closest and second closest detection $\mathbf{\Delta}$ frames away and construct the set of closest $\mathcal{C}^\star{}_{\mathbf{\Delta}}$ and second closest $\mathcal{S}^\star{}_{\mathbf{\Delta}}$ detection pairs. This procedure is summarized in Algorithm 1. These sets can then be used to learn model parameters under each model hypothesis for each feature and time interval.

**Learning from Tracklets.** The assumption that parameters can be learned from closest and second closest detections holds reasonably well for small values of $\mathbf{\Delta}$ or low crowding, but might not be verified for larger temporal gaps. However, since our tracking framework with models learned as above for relatively small $T_w$ leads to pure tracklets [18], we can use these intermediate tracklets to collect more reliable data for each hypothesis and learn more discriminative model parameters, up to a higher value of $T_w$.

[2]In principle, all non-closest detections would correspond to different persons. However, we used the second closest detection to obtain more discriminative models, esp. for the position feature.

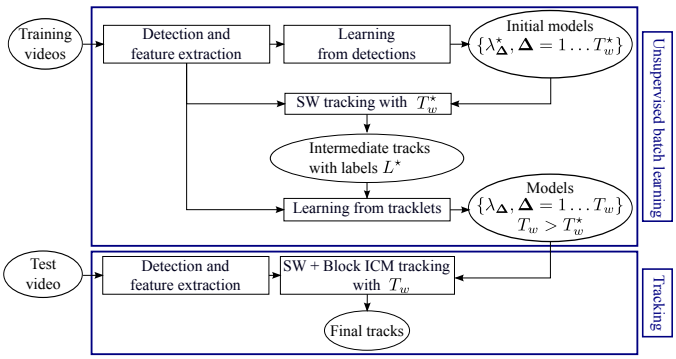


Fig. 8. Flowchart of the unsupervised batch learning and subsequent tracking procedure. Detections and features are extracted on scene-specific training videos. Initial models up to $T_w^\star$ are learned from detections. Tracking is performed with these models to obtain an intermediate labelling $L^\star$, which is in turn used to relearn more accurate models up to $T_w > T_w^\star$. Finally, given detections and features of a test video, these refined models for the scene are used to perform tracking (SW and Block ICM are two tracking steps that will be explained in Section VII).

**Algorithm 1** Collection of $\mathcal{C}^\star{}_{\mathbf{\Delta}}$ and $\mathcal{S}^\star{}_{\mathbf{\Delta}}$ from detections.

**for** $\mathbf{\Delta} = 1$ to $T_w^\star$ **do**
  Initialize empty sets $\mathcal{C}^\star{}_{\mathbf{\Delta}}$ and $\mathcal{S}^\star{}_{\mathbf{\Delta}}$
  **for** $i = 1$ to $N_r$ **do**
    $j = \arg\min_k |\mathbf{x}_i - \mathbf{x}_k|,\ s.t.\ |t_k - t_i| = \mathbf{\Delta}$
    $m = \arg\min_k |\mathbf{x}_i - \mathbf{x}_k|,\ s.t.\ t_k = t_j\ and\ k \neq j$
    Add pair $(r_i,r_j)$ to set $\mathcal{C}^\star{}_{\mathbf{\Delta}}$ and pair $(r_i,r_m)$ to set $\mathcal{S}^\star{}_{\mathbf{\Delta}}$
  **end for**
**end for**

This is illustrated in Fig. 9 for the torso color model. We can observe that for small time gaps ($\mathbf{\Delta} = 1$) the Bhattacharyya distance distributions are well separated under the two hypotheses, even when using the raw detections. However, as $T_w$ increases (e.g. for $\mathbf{\Delta} = 15$), the collected feature sets $\mathcal{C}^\star{}_{\mathbf{\Delta}}$ and $\mathcal{S}^\star{}_{\mathbf{\Delta}}$ from the detections do not correspond to the assumption any more and become more blended w.r.t. the $H_1$ or $H_0$ hypothesis, resulting in non-discriminant parameter estimates. Instead, we propose to collect new sets $\mathcal{C}_{\mathbf{\Delta}}$ and $\mathcal{S}_{\mathbf{\Delta}}$ of detection pairs for learning, using the intermediate track information, i.e. the current labelling $L^\star$. The procedure of collecting these sets from tracklets is summarized in Algorithm 2. When using the tracking results obtained with $T_w^\star = 8$ (and model parameters learned from the raw detections) to collect training data, we obtain more accurate and sensible (and still discriminative) distributions, especially for large values of $T_w$. Note that the method is unsupervised and the relearned models are still global (i.e. not specific to any track or detection).

**Robust estimates.** The above approach assumes that we obtain representative training sets for both hypotheses. While this might be true for the dissimilar hypothesis $H_0$, we actually miss large measurements for the similar case $H_1$, since tracks might actually be broken (fragmented) at places with

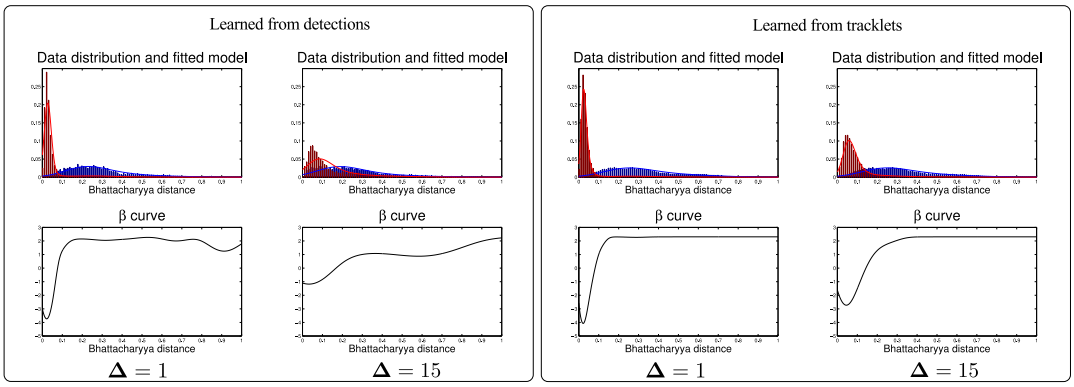


Fig. 9. Unsupervised color parameter learning. 1st row: Torso pairwise feature (color Bhattacharyya distance) distribution fitted on actual PETS data under the $H_1$ hypothesis, i.e. labels are supposed to be the same (red curve), and $H_0$ (blue curve), for two different values of $\mathbf{\Delta}$, relying on feature training sets collected from raw detections (left) and from tracklets (right). 2nd row: Corresponding $\beta$ curves of color model.

**Algorithm 2** Collection of $\mathcal{C}_{\mathbf{\Delta}}$ and $\mathcal{S}_{\mathbf{\Delta}}$ from intermediate labelling $L^\star$.

```
for Δ = 1 to T_w do
  Initialize empty sets C_Δ and S_Δ
  for each unique label l ∈ L* do
    for (r_i, r_j) ∈ τ_l with |t_i − t_j| = Δ do
      Add pair (r_i, r_j) to set C_Δ
    end for
  end for
  for each pair of unique labels (l, l') with l ≠ l' do
    for r_m ∈ τ_l and r_n ∈ τ_l' with |t_m − t_n| = Δ do
      Add pair (r_m, r_n) to set S_Δ
    end for
  end for
end for
```

high feature distances, and lead to an overconfident model for $H_1$. We alleviated this issue as follows. Let us denote by $\hat{p}(f_k|H_h, \lambda^k_{\mathbf{\Delta}})$ the feature distributions learned using the training sets collected as above. Then, we used:

$$p(f_k|H_1, \lambda^k_{\mathbf{\Delta}}) = 0.9\hat{p}(f_k|H_1, \lambda^k_{\mathbf{\Delta}}) + 0.1\hat{p}(f_k|H_0, \lambda^k_{\mathbf{\Delta}}) \quad (21)$$

and $p(f_k|H_0, \lambda^k_{\mathbf{\Delta}}) = \hat{p}(f_k|H_0, \lambda^k_{\mathbf{\Delta}})$ as actual feature distributions in the tracking framework. Intuitively, the above heuristic implicitly assumes that some measurements in the $H_0$ training set are actually coming from the same person tracks and thus should be incorporated in the $H_1$ distribution. In practice it leads to the saturation effect shown on $\beta$ curves.

## VII. Optimization

We formulated multi-person tracking as the minimization of the energy function presented in Eq. 7. The energy is decomposed into two components, the sum of feature-specific pairwise terms (Potts coefficients) weighted by their confidence, and higher-order cost terms (label costs).

Although our energy (dropping the high-order term) expression looks like a standard pairwise optimization problem, it can be shown (see supplementary material) that it does not follow the submodularity principle and hence can not be solved using global graph cut optimization techniques [21]. Instead, we introduce an iterative approximate algorithm to find a good labeling solution. More precisely, we start the labeling process by applying a Sliding Window approach. Then, in a second step we perform a more global block Iterated Conditional Modes (ICM) optimization. The two steps are summarized below, and details are provided in supplementary material.

**Sliding Window (SW).** The first step performs the labeling of the incoming detections at time step $t$ given the links with the past detections, and is formulated as an assignment problem. Essentially, each new detection can either extend an existing track $l$ (i.e. a track that has at least one detection with this label within the $T_w$ past instants), or start a new track, while existing tracks are either extended or stopped. An association matrix $\mathbf{A}^{SW}$ is thus constructed, whose elements only account for the energy terms that are affected by the assignment, which is solved using the Hungarian algorithm. Note that at this point, the higher-order label costs are not used, since we do not want to penalize ending old tracks or starting new ones to avoid initial identity switches. As shown in the results, this SW optimization already leads to very good results given the use of the long-term connectivities, and generally produces pure but fragmented tracks.

**Block ICM optimization.** In this step, optimization is conducted at a more global level and includes the higher-order label costs. The procedure is similar to [11] and is illustrated in Fig. 10. For a given time $t$, the current labeling is used to create the set of $N^B_\tau$ tracks existing *before* $t$ and the set of $N^A_\tau$ tracks existing *after* (and including) $t$.[3] Then, as with SW, the labeling is formulated as an assignment problem, where past tracks can be extended or stopped, and future tracks can extend a track or start a new one. An association matrix $\mathbf{A}^{BI}$ is built such that it comprises all terms that depend on the assignment: the pairwise terms, which involve only links within a temporal neighborhood of $T_w$ around $t$ (hence the block ICM terminology), and the global start and end label cost terms. The optimal assignment is solved with the Hungarian algorithm. The procedure is repeated for each time $t$, and in practice, we notice that one or two sweeps over all time instants $t$ are sufficient to reach the optimum.

**Optimization and tracking summary.** Optimization for

[3]Note that this may involve splitting a current track that extends before and after $t$ in two parts.

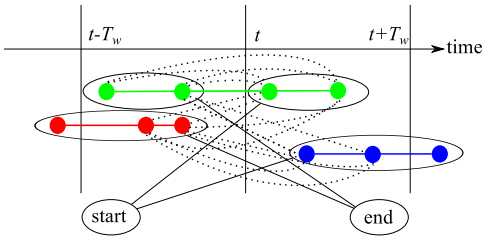


Fig. 10. Block ICM at time $t$. Current tracks before and after $t$ are associated so as to minimize block-wise $\beta$ costs between pairs (dotted lines) and label costs related to the start and ending of tracks.

tracking thus consists of two steps. First, SW is applied. It is an online procedure that labels the detections of the current frame given a set of previously labeled tracks within a sliding window, end therefore does not correct the labels of other detections within the sliding window. Even with this limitation, experiments show that SW produces pure results thanks to long-term connectivities with the past.

When handling test sequences in batch mode (as done in evaluation), the SW step is applied until it reaches the end of the sequence. Then, in a second step, Block ICM is applied, considering the whole sequence and deciding for each pair of successive frames to continue, stop or start tracks, as described above. Block ICM is therefore able to correct mistakes done at the SW level, due to its use of label costs and of both past and future observations at a given frame.

When exploited in an online system processing incoming video streams, the above strategy could be adapted. For instance, SW could be applied at every frame (using a sliding window size $T_w$ of typically a few seconds), while Block ICM could be invoked from time to time to correct SW labelling within a larger sliding window (typically about 10 seconds). In that case, $t_0$ of Section V would refer to the start of this larger sliding window used by Block ICM [4], while $t_{end}$ would be the end of this larger window, i.e. would correspond to the latest available frame of the video stream.

## VIII. Experiments

We conducted experiments on five different datasets, described in Section VIII-A. Experimental details are given in Section VIII-B. Performance metrics are presented in Section VIII-C. Section VIII-D first presents the impact and benefit of the different modeling contributions, while comparisons against state-of-the-art methods as well as qualitative tracking results are given in Section VIII-E. Finally, Section VIII-F provides some discussion on the algorithm complexity.

### A. Datasets

We used five public datasets for which bounding box annotations are available (see samples frames in Fig. 13 to 17). For all datasets, unless specifically mentioned, we are using the official ground truth files.

**PETS 2009.** PETS'09 S2.L1 [1] is a video of 795 frames recorded at 7 fps. It presents a moderately crowded scene where 20 pedestrians are often crossing each other's trajectories, creating inter-person occlusions. People are also often occluded by a street light in the middle of the scene, creating miss-detections. Although several views of the same scenario are available, we are working solely in View 001. As there is no official ground truth available for PETS, we are using the one provided by [29].

**TUD.** It consists of three short videos recorded at 25 fps. We focus on the two longest ones, which are also the ones presenting the most occlusions: TUD-Crossing (201 frames, 13 pedestrians) and TUD-Stadtmitte (179 frames, 10 pedestrians), showing respectively a pedestrian crossing and a town-centre pedestrian area. These videos have a low view point, on the contrary to the PETS sequence.

**CAVIAR.** This corpus contains 26 monocular videos of a corridor view recorded at 25 fps [2]. The average video length is 1500 frames. To compare our performance to competitive approaches, we use the same subset of 20 videos as [40], [19], containing 140 people, along with their selected ground truth, in which fewer persons are annotated as compared to the complete CAVIAR ground truth. Challenges in this dataset arise from reflections on the floor, projected shadows, occlusions, and numerous possible entry and exit points.

**Parking Lot.** The Parking Lot dataset used by [39] is a 1000-frame video recorded at 29 fps, containing 14 pedestrians walking in queues. Challenges in this dataset include long-term inter-object occlusions, and appearance similarities between several subjects.

**Town Centre.** The Town Centre dataset introduced by [6] is a high-definition surveillance video of a busy town centre street recorded at 25 fps. This dataset is challenging because it contains a large number of people frequently occluding each other. Bounding box annotations are given for 3 minutes of this video.

### B. Experimental details and parameters

**Human detection.** In tracking-by-detection approaches, the tracking performance is subject to the detection accuracy. In the literature, different authors often apply different detectors suited to their techniques on a given dataset. For instance, on the PETS dataset, Ben Shitrit et. al. [29] use the POM detector [16] which exploits multi-camera information, Breitenstein et. al. [10] use the HOG detector [12], Andriyenko et. al [4] use a detector exploiting both HOG [12] and relative optical flow (HOF) [32] features within SVM classification. Similarly to us, Zamir et. al. [39] use the part-based model detector [15]. Hence, it is currently very difficult to have fair comparisons by re-using available detection results, as pointed out in [25].

In this work, the entry to our tracking-by-detection framework is the output of the part-based detector [15] using the human deformable model trained on the INRIA person dataset [20]. As mentioned in Section III-A, this is an algorithmic choice allowing to extract motion from discriminatively trained parts (details given below) and to be able to use the same detector and method for all experiments and all datasets. It also presents the advantage of relying on a publicly available

[4]Or the corresponding effect could be neglected since the start of the video is far in the past.

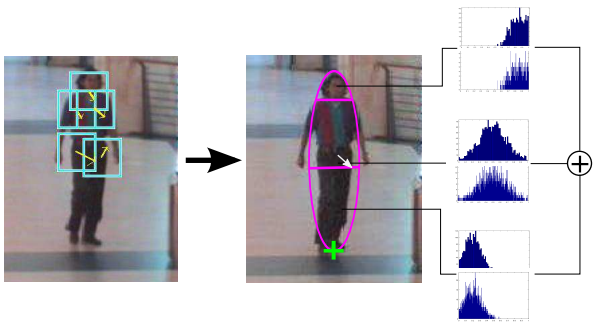

Fig. 11. Extracted features for representing detections. Left: upper-body parts obtained from the deformable parts model (cyan bounding boxes) and estimated motion on each part (yellow arrows). Right: position (green cross), final motion feature (white arrow) and color histograms obtained from different pre-defined parts (head, torso, legs and fullbody).

detector. Note that the part-based model detector that we use does not completely solve the detection problem by itself. Indeed, as shown in Tables V and IX, our detector gives us similar input detection accuracies as compared to other approaches. Hence, the results shown in the manuscript are based on input detections that are affected by severe occlusions, false positives and misses.

**Motion computation.** Several techniques could be applied to extract the motion vector $\mathbf{v}_i$ of a detection $r_i$. In this work, it is extracted by estimating an affine motion model on each of the 5 upper-body parts of the deformable part model (see Fig. 11) using the robust multi-resolution approach by [26], which provides individual part motion along with a confidence weight (as explained in Section III-A). The overall motion is then obtained as the weighted average of these upper-body parts motions. Note that these upper body parts are not the limbs, but the head, shoulders and lower torso. We observed that their motion is in general similar. Confidence weights given by [26] contribute to lower the scores of parts with unreliable motion.

**Color histograms.** To avoid taking into account too many pixels from the background, we only consider the elliptical region enclosed within each bounding box. The parts are defined by vertically partitioning the ellipse into three parts, with the top 20% aiming at capturing the head, the 40% and 40% left in the middle and the bottom aiming at capturing the torso and the legs, respectively, as illustrated in Fig. 11. As color descriptors $\mathbf{h}_i^b$ for each of the 4 pre-defined parts $b \in \mathcal{P} = \{\text{whole}, \text{head}, \text{torso}, \text{legs}\}$, we used RGB multi-resolution histograms (at resolutions $4 \times 4 \times 4$ and $8 \times 8 \times 8$) to reduce quantization effects.

**Calibration.** For the PETS and CAVIAR datasets, camera calibration and ground-plane homography are available, respectively. Using this information, position models are defined in the ground plane. On the other hand, the position models for the TUD dataset are defined in the image plane.

**Parameters.** Besides $\lambda$ which are learned automatically, the same following parameters were used in all sequences: $\theta_f = 10$ frames for the position model forgetting factor (Section IV-E); $d_{\max} = 10$ frames and $\theta_p = 3$ frames to define the label cost (Section V). Besides, unless stated otherwise, unsupervised learning of interval sensitive parameters from tracklets was conducted, all features (including motion) were used, and SW optimization followed by block ICM exploiting label cost with $\rho = 1$ was applied. Finally, we vary the size $T_w$ of the temporal window to analyze the impact of connectivity.

### C. Evaluation Metrics

In multiple person tracking literature, different existing evaluation metrics are not consistently used by competing approaches [25]. To achieve a fairer comparison with existing approaches, we use two types of measures to perform our evaluations. Measures introduced in [23] indicate how correct the tracks are in terms of fragmentation and confusion between different people. Namely, Frag is the number of times that a ground truth trajectory is interrupted in the tracking result, while IDS is the total number of identity switches, i.e. it indicates the number of times an output track is associated to several ground truth targets.

In order to compare our input detections to the ones used by other authors, when available, we report Det. Prec. and Det. Rec, which are respectively the frame-based precision and recall of the raw detections. The precision is defined as the number of correctly matched detections over the total number of detection outputs. The recall is defined as the number of correctly matched detections over the total number of ground-truth objects. On all datasets, these measures are computed following the VOC criterion, with a threshold of 0.2 on the intersection over union for matching. We also provide recall and precision after tracking (Rec. and Prec.) by using tracking information to interpolate tracks and remove short ones.

Finally we report the number of tracker outputs SO, the percentage of tracks that are tracked for more than 80% of their duration MT (Mostly Tracked), the percentage of tracks that are tracked between 20% and 80% of their duration PT (Partially Tracked) and the percentage of tracks that are tracked less than 20% of their duration ML (Mostly Lost).

Since the above metrics are not adopted by several competing sate-of-the-art tracking methods, we additionally use the CLEAR MOT metrics MOTA and MOTP [9]. "Multi-Object Tracking Accuracy" (MOTA) combines missed detections, false positives and identity switches into a single evaluation measure. On the other hand, "Multi-Object Tracking Precision" (MOTP) gives a measure on bounding boxes localization accuracy.

### D. Results and evaluation of different modeling factors.

Sample tracking outputs can be seen in Fig. 13 to 17. In the following, we demonstrate quantitatively the benefit of the different modeling factors on the results. Note that more thorough results supporting the different claims are provided in the Supplementary material for the PETS, TUD, and CAVIAR datasets.

**Unsupervised learning.** Table I demonstrates the effect of learning model parameters from tracklets rather than from detections, as explained in Section VI. In practice, we used tracklets obtained with models learned from detections with $T_w = 8$ (first line of Table I) to relearn models from tracklets up to $T_w = 16$. We can observe that the refinement of model parameters using tracklets has almost no effect on the

| $T_w$ | *MET* | Rec | Prec | SO | MT | PT | ML | Frag | IDS |
|---|---|---|---|---|---|---|---|---|---|
| 8 | Off | 0.84 | 0.95 | 40 | 70% | 25% | 5% | 13 | 1 |
| 8 | On | 0.84 | 0.95 | 39 | 70% | 25% | 5% | 12 | 0 |
| 16 | Off | 0.82 | 0.95 | 92 | 60% | 35% | 5% | 27 | 0 |
| 16 | On | 0.87 | 0.94 | 25 | 70% | 25% | 5% | 3 | 0 |

TABLE I
UNSUPERVISED LEARNING. SW OPTIMIZATION FOR PETS USING MODEL PARAMETERS ESTIMATED FROM TRACKLETS ($MET$ = "ON"), OR NOT ($MET$ = "OFF").

| $T_w$ | *TW* | Rec | Prec | SO | MT | PT | ML | Frag | IDS |
|---|---|---|---|---|---|---|---|---|---|
| 16 | Off | 0.86 | 0.94 | 26 | 70% | 25% | 5% | 6 | 3 |
| 16 | On | 0.87 | 0.94 | 25 | 70% | 25% | 5% | 3 | 0 |

TABLE II
SW OPTIMIZATION OUTPUT FOR PETS SEQUENCE USING TIME-INTERVAL SENSITIVE MODELS ($TW$ = "ON") OR NOT ($TW$ = "OFF") FOR THE COLOR AND MOTION MODELS.

performance for $T_w = 8$, showing that the assumption of using the closest and second closest sets of detection pairs to learn models is valid for small values of $T_w$. However, with a larger association window ($T_w = 16$), using the default models leads to precise but very fragmented tracklets (92 different labels, 27 Frag). This fragmentation can be dramatically reduced by using the refined parameter estimates obtained from tracklets, showing the benefit and validity of our approach. We observed the same benefit of learning from tracklets on other datasets.

**Time interval sensitivity.** One might argue that learning motion and color similarity models that depend on the time gap between detection pairs may have no impact on the results, since within our association windows, motion and appearance patterns of an individual are likely to stay similar. However, Table II demonstrates empirically that exploiting such time-interval dependent models indeed helps reaching better tracking performance, and confirms the dependencies observed on the learned $\beta$ curves (see Fig. 5 and 9). When the motion and color features between pairs of detections are collected from tracklets regardless of their time difference ($TW = 0$), worse results are obtained (the position model is learned normally), resulting in 3 more fragmentations and IDS. A similar behavior has been observed on the other datasets.

**Temporal context.** The benefit of using a longer temporal connectivity between detection pairs is demonstrated in Table III, where we observe that larger $T_w$ values reduce fragmentations. This is due to two main reasons. First, note that tracks for which there are long intervals with no detections (beyond $T_w$) can not receive the same label, since no link is created between the detections before and after the miss-detection interval. Hence, increasing $T_w$ can solve these miss-detection and occlusion situations. This is mainly illustrated in PETS where people tend to get occluded by the street lamp for more than 10 frames. By increasing $T_w$ to a value of 16, the number of fragmentations gets significantly reduced (e.g. from 12 to 3 when using all features). The second reason is that a longer temporal connectivity that relies on all pairwise links leads to an energy that is better conditioned for optimization, or in other words, that provides a better temporal context for labeling. This is illustrated in Fig. 12 in an example from TUD-Crossing.

**Visual motion cue.** Table III also demonstrates the usefulness of the motion feature at solving ambiguities and therefore reducing the number of identity switches. In practice, these

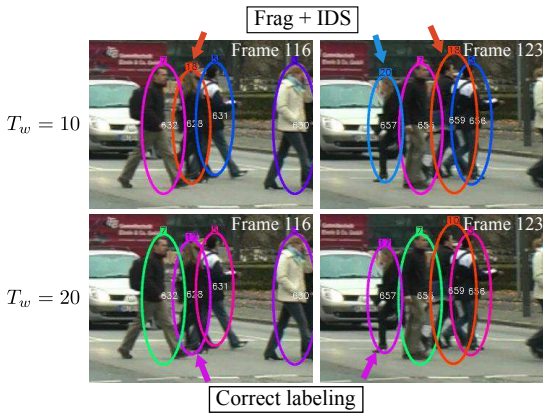


Fig. 12. Temporal context effect. First row: Even though the occluded person with the orange label (#18) reappears less than $T_w = 10$ frames later, the links do not provide enough context to reassign her with the correct label. Bottom row: when a longer context is available ($T_w = 20$) more pairwise comparisons are available, allowing to maintain a correct labeling.

| | $T_w$ | *motion* | Rec | Prec | SO | MT | PT | ML | Frag | IDS |
|---|---|---|---|---|---|---|---|---|---|---|
| PETS | 8 | Off | 0.84 | 0.95 | 38 | 70% | 25% | 5% | 13 | 2 |
| | 8 | On | 0.84 | 0.95 | 39 | 70% | 25% | 5% | 12 | 0 |
| | 16 | Off | 0.87 | 0.94 | 23 | 70% | 25% | 5% | 4 | 3 |
| | 16 | On | 0.87 | 0.94 | 25 | 70% | 25% | 5% | 3 | 0 |
| TUD | 10 | Off | 0.77 | 0.98 | 20 | 70% | 30% | 0% | 6 | 2 |
| | 10 | On | 0.77 | 0.99 | 20 | 70% | 30% | 0% | 6 | 2 |
| Stadtmitte | 20 | Off | 0.79 | 0.98 | 19 | 70% | 30% | 0% | 5 | 2 |
| | 20 | On | 0.79 | 0.99 | 19 | 70% | 30% | 0% | 4 | 1 |

TABLE III
RESULTS ON PETS AND TUD-STADTMITTE SEQUENCES WITH SLIDING WINDOW OPTIMIZATION. USING THE MOTION FEATURE (MOTION="ON") AND LARGER TEMPORAL WINDOW $T_w$ PROVIDES BETTER RESULTS.

ambiguities happen mainly when people with similar appearance are crossing trajectories and there are important missdetection periods and badly framed detections (i.e. encapsulating parts of the two people). The position model that does not favor any movement direction and the color model might not be discriminant enough to solve the association in these cases, and the motion feature adds the complementary information. Note here that confidence weighting is important, as motion estimates at the near proximity of the crossing might be unreliable because bounding boxes tend to get blended together, but previous motion estimates are then prevailing in the energy term because of their higher confidence (the same goes for the color models). In the end, by using the motion feature and a sufficiently large value of $T_w$, we are able to obtain pure tracklets with few IDS in general.

**Label costs and block ICM optimization.** We evaluated the benefit of using label costs with a more global optimization to improve performance. On PETS data, where the Sliding Window approach already provides very good tracking results with only 3 fragmentation and 0 IDS, no improvement was observed. However, results on TUD-Stadtmitte with $\rho = 3$ (Table IV) shows that several errors can be corrected, allowing us to reach a very good performance of just 1 Frag and 0 IDS. Similar benefit could be observed on CAVIAR data, where block ICM and label cost acted towards fragmentation reduction while solving some IDS ambiguities as well.

From our experiences, it stands out that block ICM with label costs can be useful to correct some mistakes through the incorporation of track start and end penalizations leveraging

| $T_w$ | *BlockICM* | Rec | Prec | SO | MT | PT | ML | Frag | IDS |
|---|---|---|---|---|---|---|---|---|---|
| 20 | Off | 0.79 | 0.99 | 19 | 70% | 30% | 0% | 4 | 1 |
| 20 | On | 0.81 | 0.99 | 18 | 70% | 30% | 0% | 1 | 0 |

TABLE IV
EFFECT OF BLOCK ICM WITH LABEL COSTS FOR TUD-STADTMITTE.

| | Det. Rec | Det. Prec | Rec | Prec | Frag | IDS |
|---|---|---|---|---|---|---|
| Huang et. al. [19] | 0.88 | 0.70 | 0.86 | - | 54 | 12 |
| Zhang et. al. [40] algo 1 | 0.88 | 0.70 | - | - | 58 | **7** |
| Zhang et. al. [40] algo 2 | 0.88 | 0.70 | - | - | **20** | 15 |
| **Ours** | 0.82 | 0.69 | 0.78 | 0.93 | 38 | 8 |

TABLE V
COMPARISON WITH STATE OF THE ART APPROACHES ON CAVIAR.

| | Rec | Prec | Frag | IDS | MOTA | MOTP |
|---|---|---|---|---|---|---|
| Breitenstein et. al. [10] | - | - | - | 2 | 0.84 | 0.71 |
| Zamir et. al. [39] | 0.93 | 0.99 | - | **0** | **0.92** | 0.76 |
| **Ours** | 0.89 | 0.93 | **1** | **0** | 0.79 | **0.78** |

TABLE VI
COMPARISON WITH STATE-OF-THE-ART APPROACHES ON TUD-CROSSING.

on scene-specific knowledge to define prior label information.

### *E. Comparison with the state-of-the-art.*

Tables V, VI, VII, VIII and IX show the comparison with recent state-of-the-art algorithms for the different datasets, when available. Although there are public methods for tracking evaluation, there is a lack of a unique standard procedure (i.e, some authors use MOT metrics while others use fragmentation and IDS). This makes fair comparison against several methods, including recent ones, difficult, as pointed by Milan et. al. [25]. In this paper, we evaluate our performance with different existing metrics to allow comparison with existing approaches that have some similarities to our proposal. Note as well that, as discussed in Section VIII-B, different authors often use different detectors. For the sake of having more detailed comparisons, we also report and discuss the input detection recall and precision of our detections and compare them to those of the detections provided by the different authors, when available.

On the CAVIAR dataset, Table V compares our results obtained with an association horizon of 1.5 second ($T_w = 38$) and default parameters, with the approaches from [19] and [40]. Note first that our detector delivers lower performance, with a worse detection recall for a comparable detection precision. Nevertheless, the table shows that we outperform [19] in terms of Frag and IDS. As compared to the network flow formulation of [40] (algo. 1), we reach an almost identical number of IDS (8 vs. 7) but with much less fragmented tracks (38 vs. 58). When adding an explicit occlusion model on top of the flow model (algo. 2), the method in [40] reduces the number of fragmentations to 20, but this is at the cost of a higher number of IDS (15). Our approach thus offers a good tradeoff between their methods.

For the TUD and PETS datasets, we report our results obtained with $T_w = 20$ and $T_w = 16$, respectively. In the TUD-Crossing sequence which contains heavy occlusions, we obtain 1 Frag and 0 IDS, outperforming the method of [10] (2 IDS) and we equal [39] in terms of IDS. However, they both present a better MOTA score. This can be explained by the fact that MOTA takes into account not only IDS, but also tracking precision and recall. In this sequence, people are often occluded because they walk next to each other, and this translates into low detection recall. For instance, by the end of the sequence we miss a subject due to such an occlusion, because we did not get any detection in the first place. Since the proposed method does not attempt to propagate detections nor extrapolate tracklets, such missdetections penalize the tracking recall, and ultimately the MOTA. The methods of [10] and [39] generate candidate detections by using particles and virtual nodes, respectively, potentially overcoming problems with missing detections due to occlusion. Despite the lack of detections, in this sequence our method obtains pure tracklets, with only 1 fragmentation.

On TUD-Stadtmitte, we outperform [4] both in terms of Frag, IDS and MOT metrics. We reach similar results as [39] and [36], with 1 Frag and 0 IDS. However, we outperform [39] in terms of MOT metrics.

On PETS, we clearly outperform other techniques insofar as we reach 0 IDS. The authors of [39] obtain comparable MOT metrics but with a much higher number of 8 IDS. It can be noted that one of our fragmentations is due to the fact that a person going out of the scene and coming back later is annotated as one single ground truth object. This situation is out of the scope of this paper, as we do not tackle the re-identification problem. Another fragmentation is due to a very long occlusion by the street lamp (more than 10 seconds).

Finally, we compare our tracking results to state-of-the-art methods on Parking Lot and Town Centre. On both sequences, we use a temporal connectivity of $T_w = 40$. These results are summarized in Table IX. We obtain a similar MOTA than [39] on the Parking Lot sequence. However, our tracking precision is higher. On the Town Centre sequence, we outperform [6] and [39] both in terms of MOTA and MOTP. Note that on these datasets, the recall and precision of our detections are similar to those of the detections provided by the authors of [39] for Parking Lot, and [6] for Town Centre[5].

**Qualitative results.** Finally, Fig. 13, 14 15, 16 and 17 show some visual results of our tracker on the different datasets. It can be seen that even in the presence of multiple occlusions and ambiguities, our algorithm is able to maintain correct tracks throughout time. Tracking videos are made available as supplementary material and can also be found online [6].

### *F. Computational complexity.*

Detection-based tracking approaches can basically be described as two processing steps: detection and association. With regard to the detections, although we have used the version of Felzenswalb [15] in our experiments, the DPM detection could be made faster by relying on a recently proposed accelerated version of DPM [14] which exploits Fast Fourier transform to speed up the per-part convolutions required by the algorithm. Benchmarked on the VOC dataset,

[5]We recall that Zamir et. al. [39] also use the part-based model detector on all datasets.

[6]www.idiap.ch/~aheili/tracking.html

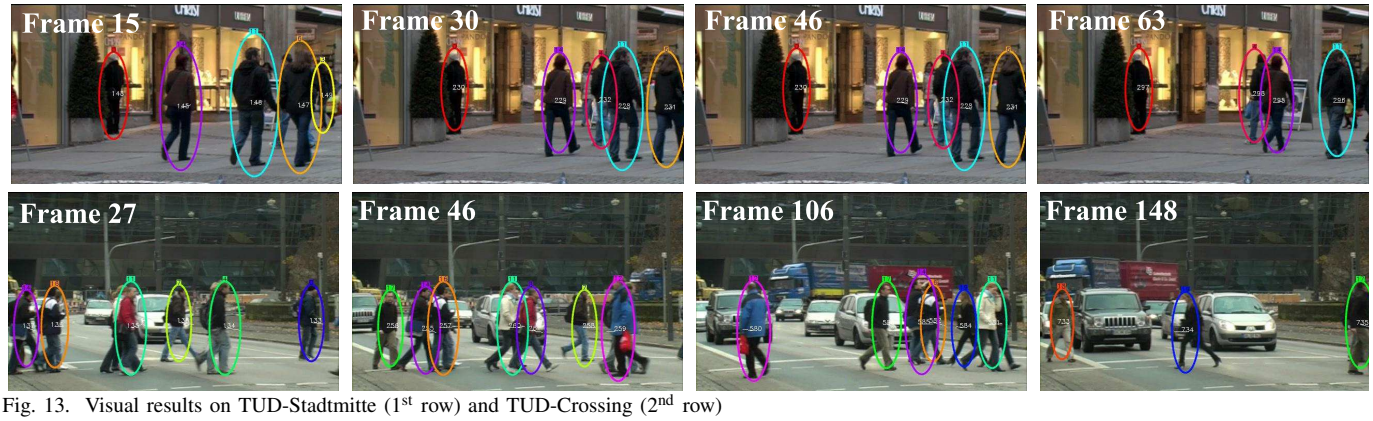


Fig. 13. Visual results on TUD-Stadtmitte (1st row) and TUD-Crossing (2nd row)

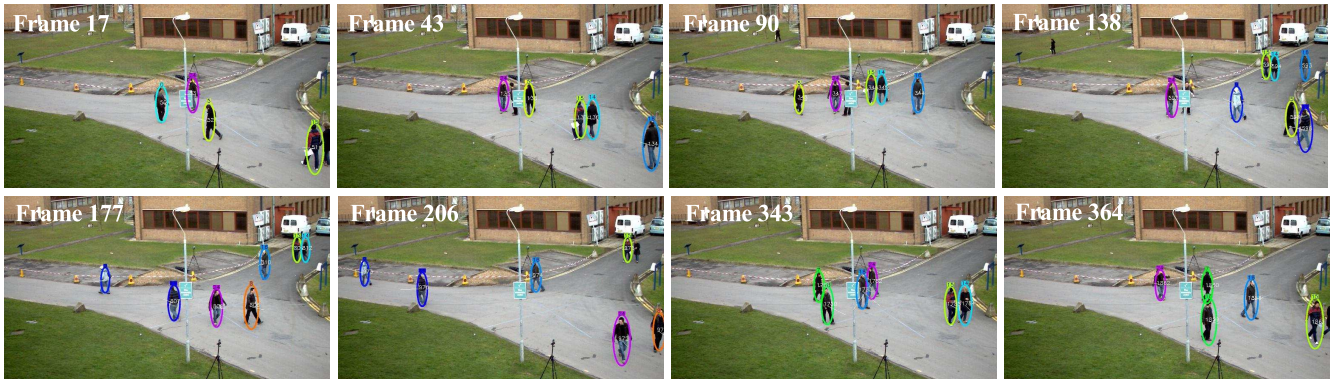


Fig. 14. Visual results on PETS S2.L1 sequence (View 001)

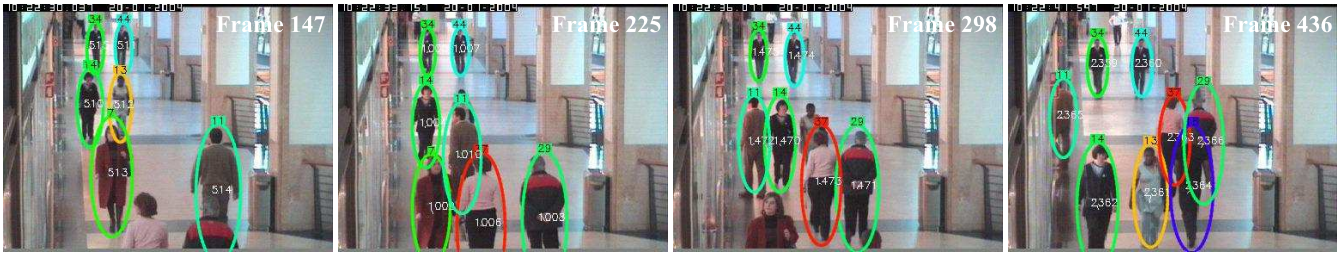


Fig. 15. Visual results on CAVIAR

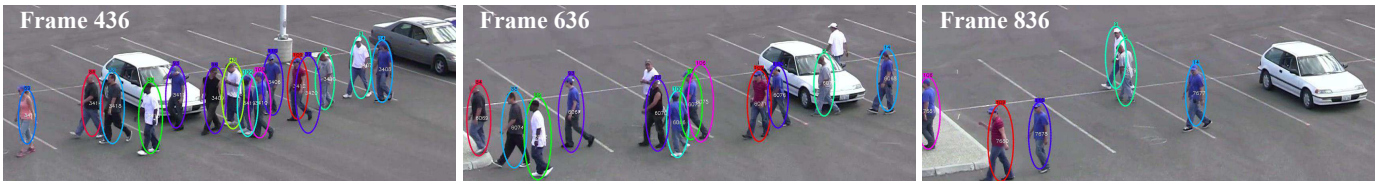


Fig. 16. Visual results on Parking Lot (images were edited to highlight interesting regions)

| | Rec | Prec | Frag | IDS | MOTA | MOTP |
|---|---|---|---|---|---|---|
| Andriyenko et. al. [4] | - | - | **1** | 4 | 0.62 | 0.63 |
| Yang et. al. [36] | 0.87 | 0.97 | **1** | **0** | - | - |
| Zamir et. al. [39] | 0.81 | 0.96 | - | **0** | 0.78 | 0.63 |
| **Ours** | 0.81 | 0.99 | **1** | **0** | **0.90** | **0.84** |

TABLE VII
COMPARISON WITH STATE-OF-THE-ART APPROACHES ON TUD-STADTMITTE.

| | Rec | Prec | Frag | IDS | MOTA | MOTP |
|---|---|---|---|---|---|---|
| Andriyenko et. al. [4] | - | - | 8 | 10 | 0.89 | 0.56 |
| Shitrit et. al. [29] | - | - | - | 9 | - | - |
| Breitenstein et. al. [10] | - | - | - | - | 0.80 | 0.56 |
| Zamir et. al. [39] | 0.96 | 0.94 | - | 8 | **0.90** | **0.69** |
| **Ours** | 0.87 | 0.94 | **3** | **0** | 0.89 | 0.66 |

TABLE VIII
COMPARISON WITH STATE-OF-THE-ART APPROACHES ON PETS S2.L1.

| | | Det. Rec | Det. Prec | MOTA | MOTP |
|---|---|---|---|---|---|
| Parking Lot | Zamir et. al. [39] | 0.86 | 0.96 | **0.90** | 0.74 |
| | **Ours** | 0.91 | 0.96 | 0.89 | **0.85** |
| Town Centre | Benfold et. al. [6] | 0.77 | 0.88 | 0.65 | 0.80 |
| | Zamir et. al. [39] | - | - | 0.76 | 0.72 |
| | **Ours** | 0.73 | 0.90 | **0.79** | **0.82** |

TABLE IX
COMPARISON WITH STATE-OF-THE-ART APPROACHES ON PARKING LOT AND TOWN CENTRE.

the algorithm was shown to provide a speed-up of one order of magnitude over the DPM baseline.

While the main cost of the human detector is very proportional to the size of the input image and does not depend on its content, the tracker cost can arise from the appearance information extraction, the graph link construction, and the graph optimization. Appearance feature extraction is done once for every detection, and is thus not affected by the amount


Fig. 17. Visual results on Town Centre (images were cropped to save space)

of temporal connectivity. Pairwise $\beta$ term computation to build the graph, however, depends directly on the connectivity, but relies on simple distances between feature vectors whose computation cost is small or that can be easily optimized.

As for the optimization, since the SW algorithm depends on the Hungarian algorithm, its complexity is polynomial in $O(n^3)$, where $n$ is the maximum between the number of detections in the current frame and the number of current tracks in the sliding window. Therefore, longer term connectivity does not necessarily imply an increase in complexity. Indeed, as there are typically fewer fragmentations (and thus less tracks) when using longer temporal windows, the complexity might even be reduced. Similarly, block ICM is optimized using the Hungarian algorithm, and its complexity is polynomial in the maximum of the number of tracks before and after the currently optimized frame in the ICM sweep.

To give an idea about the computational complexity of our tracking algorithm, we report the following average processing times per frame on the medium crowded scene of PETS 2009 with an association horizon $T_w$ of 2 seconds, tested on a 2.9 GHz Intel Core i7 laptop with 8GB of RAM and assuming detections are available: 150ms for visual motion estimation and color features extraction; 180ms for computing the pairwise $\beta$ terms; 60ms and 280ms for SW and Block ICM optimization, respectively. Note that we have an unoptimized implementation in Python with no threading. Online tracking processing could be achieved by optimizing algorithmic steps -for instance using simple and quick procedures to trim unnecessary links in the graph[7] or the time steps at which applying Block ICM could be useful, or through code optimization (programming language, multi-threading, etc.) as well as by processing videos at a lower framerate.

## IX. Conclusion

We presented a CRF model for detection-based multi-person tracking. Contrarily to other methods, it exploits longer-term connectivities between pairs of detections. Moreover, it relies on pairwise similarity and dissimilarity factors defined at the detection level, based on position, color and also visual motion cues, along with a feature-specific factor weighting scheme that accounts for feature reliability. The model also incorporates a label field prior penalizing unrealistic solutions, leveraging on track and scene characteristics like duration and start/end zones. Experiments on public datasets and comparison with state-of-the-art approaches validated the different modeling steps, such as the use of a long time horizon $T_w$ with a higher density of connections that better constrains the models and provides more pairwise comparisons to assess the labeling, or an unsupervised learning scheme of time-interval sensitive model parameters.

There are several possibilities to extend our work. First, rather than using the same model parameters for the whole test sequence, unsupervised learning or adaptation of model parameters could be done online by considering detection outputs until the given instant while performing tracking on long videos. Second, in order to handle the high-level of miss-detections that can negatively impact our algorithm, short term forward and/or backward propagations of detections could be generated and directly used as another pairwise association cue in our framework. Furthermore, to handle long occlusions (beyond 3s and more), higher order appearance re-identification factor terms potentially relying on online learned discriminative models like [5] should be defined and exploited at another hierarchical level. Finally, to better handle crowd and small group moving interactions, high-order dynamical prior model taking into account multiple tracks jointly could be defined like in [7] and used to constrain the solution space in the global optimization stage.

[7]For example, by not creating links between detection pairs that are separated by unrealistic distances.

## Acknowledgment

This work has been funded by the Integrated Project VANAHEIM (248907) supported by the European Union under the 7th framework program.

## References

[1] http://www.cvg.rdg.ac.uk/PETS2009/.

[2] http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/.
[3] A. Andriyenko and K. Schindler. Multi-target tracking by continuous energy minimization. In *CVPR*, pages 1265–1272, 2011.
[4] A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. In *CVPR*, pages 1926–1933, 2012.
[5] S. Bak, D. P. Chau, J. Badie, E. Corvee, F. Bremond, and M. Thonnat. Multi-target tracking by discriminative analysis on Riemannian manifold. In *ICIP*, pages 1–4, 2012.
[6] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR*, pages 3457–3464, 2011.
[7] J. Berclaz, F. Fleuret, and P. Fua. Multi-camera tracking and atypical motion detection with behavioral maps. In *ECCV*, pages 112–125, 2008.
[8] J. Berclaz, F. Fleuret, and P. Fua. Multiple object tracking using flow linear programming. In *Winter-PETS*, pages 1–8, 2009.
[9] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the CLEAR MOT metrics. *J. Image Video Process.*, 2008:1:1–1:10, 2008.
[10] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(9):1820–1833, 2011.
[11] R. T. Collins. Multitarget data association with higher-order motion models. In *CVPR*, pages 1744–1751, 2012.
[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *In CVPR*, pages 886–893, 2005.
[13] P. Dollar, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *BMVC*, pages 68.1–68.11, 2010.
[14] C. Dubout and F. Fleuret. Exact acceleration of linear object detectors. In *ECCV*, pages 301–311, 2012.
[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010.
[16] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(2):267–282, February 2008.
[17] A. Heili, C. Chen, and J.-M. Odobez. Detection-based multi-human tracking using a CRF model. In *ICCV Visual Surveillance Workshop*, pages 1673–1680, 2011.
[18] A. Heili and J.-M. Odobez. Parameter estimation and contextual adaptation for a multi-object tracking CRF model. In *PETS*, pages 14–21, 2013.
[19] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, pages 788–801, 2008.
[20] http://pascal.inrialpes.fr/data/human/.
[21] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *Trans. Pattern Anal. Mach. Intell.*, 26(2):147–159, 2004.
[22] C.-H. Kuo and R. Nevatia. How does person identity recognition help multi-person tracking? In *CVPR*, pages 1217–1224, 2011.
[23] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *CVPR*, pages 2953–2960, 2009.
[24] E. Maggio and A. Cavallaro. Learning scene context for multiple object tracking. *IEEE Trans. Image Processing*, 18(8):1873–1884, 2009.
[25] A. Milan, K. Schindler, and S. Roth. Challenges of ground truth evaluation of multi-target tracking. In *Proc. of the CVPR 2013 Workshop on Ground Truth - What is a good dataset?*, 2013.
[26] J. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348 – 365, 1995.
[27] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, pages 1201–1208, 2011.
[28] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert. Density-aware person detection and tracking in crowds. In *ICCV*, pages 2423–2430, 2011.
[29] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. In *ICCV*, pages 137–144, 2011.
[30] B. Song, T.-Y. Jeng, E. Staudt, and A. K. R. Chowdhury. A stochastic graph evolution framework for robust multi-target tracking. In *ECCV*, pages 605–619, 2010.
[31] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki, and H. Zha. An online approach: Learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene. In *CVPR*, pages 739–746, 2010.
[32] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *Conf. on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, Jun. 2010.
[33] Z. Wu, T. H. Kunz, and M. Betke. Efficient track linking methods for track graphs using network-flow and set-cover techniques. In *CVPR*, pages 1185–1192, 2011.
[34] B. Yang, C. Huang, and R. Nevatia. Learning affinities and dependencies for multi-target tracking using a CRF model. In *CVPR*, pages 1233–1240, 2011.
[35] B. Yang and R. Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *CVPR*, pages 1918–1925, 2012.
[36] B. Yang and R. Nevatia. An online learned CRF model for multi-target tracking. In *CVPR*, pages 2034–2041, 2012.
[37] M. Yang, Y. Wu, and G. Hua. Context-aware visual tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(7):1195–1209, 2009.
[38] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4):13, 2006.
[39] A. R. Zamir, A. Dehghan, and M. Shah. GMCP-tracker: Global multi-object tracking using generalized minimum clique graphs. In *ECCV*, pages 343–356, 2012.
[40] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, pages 1–8, 2008.

**Alexandre Heili** graduated from Télécom Physique Strasbourg, France, in 2009 and obtained at the same time a MSc in Computer Vision and Medical Robotics from UdS - Université de Strasbourg. He is currently a PhD student at Idiap Research Institute and École Polytechnique Fédérale de Lausanne, Switzerland, and a student member of the IEEE.

**Adolfo López-Méndez** received his B.S.c. (2007) and Master degree in Signal Theory and Communications (2009) from the Technical University of Catalonia (UPC), where he obtained his Ph.D degree. During this period, he was involved in several National and European Projects (VISION, PROVEC, CHIL, ACTIBIO, FASCINATE). In 2012, he became a post-doctoral researcher at Idiap Research Institute. His research interests include tracking, action and gesture recognition, markerless motion capture and machine learning.

**Jean-Marc Odobez** (École Nationale Supérieure de Télécommunications de Bretagne (ENSTBr), France, 1990; PhD at INRIA, Rennes University, France, 1994), was associate professor in computer science at the Université du Maine, Le Mans, France, from 1996 to 2001. He is now a senior researcher at both the Idiap Research Institute and EPFL, Switzerland, where he directs the Perception and Activity Understanding team. His main areas of research are computer vision and machine learning techniques applied to multimedia content analysis, tracking, and human activity and behavior recognition. He is the author or coauthor of more than 100 papers in international journals and conferences. He is or was the principal investigator of 10 European and Swiss projects. He holds two patents on video motion analysis. He is the cofounder of the Swiss Klewel SA company active in the intelligent capture, indexing, and Web casting of multimedia conferences. He is a member of the IEEE, and associate editor of the Machine Vision and Application journal.