Tracking Multiple People under Global Appearance Constraints*

Horesh Ben Shitrit[1] Jérôme Berclaz[1] François Fleuret[1,2] Pascal Fua[1]
[1] CVLab, EPFL, Lausanne, Switzerland [2]Idiap Research Institute, Martigny, Switzerland
{horesh.benshitrit,jerome.berclaz,pascal.fua}@epfl.ch francois.fleuret@idiap.ch

Abstract

In this paper, we show that tracking multiple people whose paths may intersect can be formulated as a convex global optimization problem. Our proposed framework is designed to exploit image appearance cues to prevent identity switches. Our method is effective even when such cues are only available at distant time intervals. This is unlike many current approaches that depend on appearance being exploitable from frame to frame.

We validate our approach on three multi-camera sport and pedestrian datasets that contain long and complex sequences. Our algorithm perseveres identities better than state-of-the-art algorithms while keeping similar MOTA scores.

1. Introduction

In this paper, we address the problem of tracking multiple people whose paths may intersect over long periods of time while retaining their individual identities. We assume that a time-independent people detector is available and provides us with probabilities of presence at various possible spatial locations. Our task is therefore to link these detections into consistent trajectories.

A standard approach to doing this is to recursively track from frame to frame, which may easily lead to irrecoverable errors if a person fails to be detected in a frame or if two detections made at different times are inappropriately linked. In order to overcome this problem, the recursive tracking approach can be replaced by either Dynamic Programming [25, 8] or Linear Programming [23, 11] over batches of frames. Both methods operate on directed graphs whose nodes represent places where people have been detected. The latter tends to be more robust than the former but scales poorly for large problems and long batches. This problem can be alleviated by linking detections over a few frames into *tracklets* that become the graph nodes to be linked [20, 28, 21]. This reduces the computational complexity and increases robustness but still relies on heuristics such as introducing occlusion nodes and the proper setting of many parameters, such as those controlling arc-costs in the graph. By contrast, it has recently been shown [2] that multi-object tracking could be formulated as a global optimization problem, which can be efficiently solved using the K-Shortest Paths algorithm (KSP) [24]. The objective function is convex, and controlled by only few parameters. However, it completely ignores appearance and can produce unwarranted identity switches in complex scenes.

In this paper, we extend the approach of [2] by using *sparse* appearance information to keep track of people's identity, even when their paths come close to each other or intersect. By sparse we mean that the appearance needs only be discriminative in a very limited number of frames. For example, in the basketball sequence of Fig. 1, all teammates wear the same uniform and the numbers on the back of their shirts can only be read once in a long while. Furthermore, the appearance models are most needed when the players are bunched together. However, it is precisely then, where they are least reliable [17]. Our algorithm can disambiguate such situations using the information from temporally distant frames. This is in contrast with many state-of-the-art approaches that depend on associating appearance models across *successive* frames [11, 8, 1].

We achieve this by solving a Linear Program on a layered graph such as the one depicted by Fig. 2, which contains several grid cells at each possible spatial location, one for each possible identity group. It is much larger than the one of the original approach that contains a single layer. However, by first running the KSP method [2] on this smaller graph, we can eliminate all the nodes in which nobody is present and run our algorithm on a much reduced layered graph, thus making the problem tractable.

The contribution of this paper is therefore both a reformulation of the identity-preserving multiple target tracking problem in terms of finding the global maximum of a convex objective function, and a practical algorithm for doing so. We validate our new method on multiple datasets featuring basketball players, soccer players, and pedestrians

*This work was supported in part by the CTI project "Image Based Object Tracking and Identification in Team Sports Environments". François Fleuret was supported in part by the European Community's Seventh Framework Programme FP7 - Challenge 2 - Cognitive Systems, Interaction, Robotics - under grant agreement No 247022 - MASH.

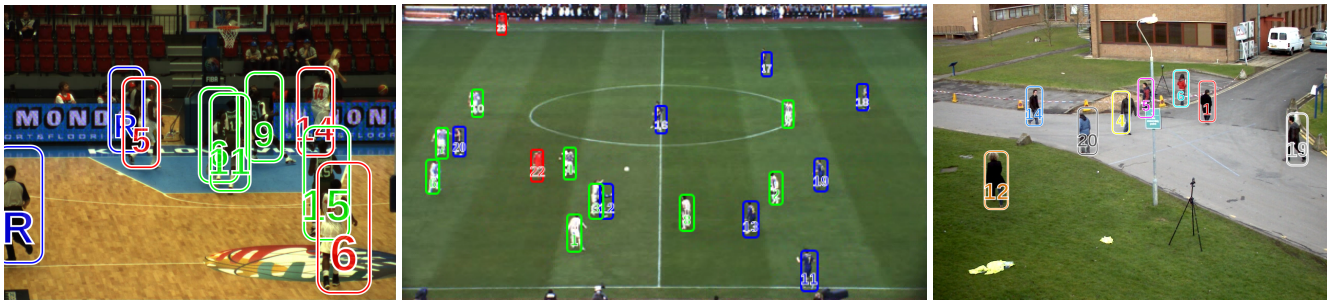


Figure 1. Representative tracking results from the three tested datasets: From left to right, Basketball, Soccer and Pedestrians.

and demonstrate a significant improvement over earlier approaches.

2. Related Work

Multiple target tracking has a long tradition, going back many years for applications such as radar tracking [5]. These early approaches to data association usually relied on gating and Kalman filtering, which have later made their way into our community [4, 18, 10, 26, 14]. Because of their recursive nature, when used to track people in crowded scenes, they are prone to identity switches that are difficult to recover from. Particle-based approaches such as [9, 22, 19, 13, 27, 16, 15], among many others, partially address this issue by simultaneously exploring multiple hypotheses. However, they can handle only relatively small batches of temporal frames without their state space becoming unmanageably large and often require careful parameter settings to converge.

In recent years, Dynamic and Linear Programming approaches have emerged as powerful alternatives. They operate on graphs whose nodes can either be all the spatial locations where somebody could potentially be [25, 8], only those where a detector has fired [23, 11], or short temporal sequences of consecutive detections that are very likely to correspond to the same person [20, 28, 21]. On average, they are much more robust than the earlier methods but typically require the careful setting of edge costs in the graph, the introduction of special purpose nodes to handle occlusions, and an assumption that the appearance of people remains both unchanged and discriminative from frame to frame. This last assumption is damaging in cases where the lighting changes quickly or where the appearance is only distinctive at long intervals, such as when tracking ballplayers who all wear the same uniform and whose number can only be read occasionally.

This limitation is entirely bypassed by a recent approach [2] that belongs to the class of those that work on the graph of all potential locations over time and solves the data association problem using the K-Shortest Paths algorithm [24]. It completely ignores appearance, does not require any heuristics regarding occlusion nodes, and has a comparatively low computation complexity in the order of $O(k(m + n \log n))$, where n, m, and k are the number of graph nodes, edges, and trajectories. And, yet, it has been shown to outperform many state-of-the-art methods on the PETS'09 database [7]. Its main limitation is that, by completely ignoring appearance, it cannot prevent identity switches when people come close to each other. This is the problem we address in this paper.

3. Algorithm

In this section, we assume that the ground plane is represented by a discrete grid and that, at each time step over a potentially long period of time, we are given as input a Probabilistic Occupancy Map [8] (POM) containing probabilities of presence of people in each grid cell, which can be generated by any people detector. While informative, the resulting probability maps may contain both missed detections and false positives, especially when the scene becomes crowded.

To infer identity-preserving trajectories from these potentially noisy POMs, we first extend the formalism introduced in [2] to account for individual identities, which the original formulation did not do. This results in our multi-target tracking problem being reformulated as an Integer Programming problem, which can be relaxed into a Linear Program. It can be solved using standard optimization packages but may be still very slow for long sequences. We therefore obtain an approximated solution much faster using a two step process: We first run the K-Shortest Paths algorithm (KSP) as in [2] to find trajectories that may include identity switches but tell us which are the grid cells in which we can expect to find people at any given time. We then run our Linear Program on a significantly reduced number of grid cells, which saves both time and memory.

3.1. Formulation

As in [2], we model people's trajectories as continuous flows going through an area of interest. Preserving identity means that available appearance cues should be used to assign it and that the flows should not be allowed to mix, two elements that are missing from the original formulation.

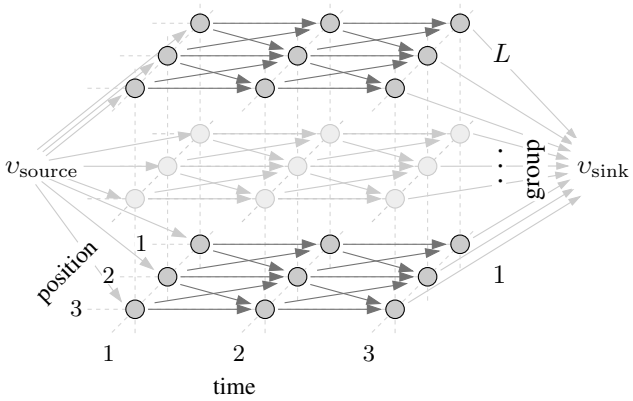


Figure 2. Our tracking algorithm is formulated using a Direct Acyclic Graph with a separate layer for each identity group. It includes source and sink node that allow people to enter and exit at selected locations, such as the boundaries of the playing field.

To this end, we discretize the ground plane into a grid containing K cells and compute POMs at T consecutive instants. We partition the total number of tracked people into L groups and assign a separate appearance to each group. In a constrained scene, such as a ball game, we can restrict each group l to include at most N_l people, but in general cases, N_l is left unbounded. The groups can be made of individual people, in which case $N_l = 1$. They can also be composed of several people that share a common appearance, such as members of the same team or referees.

We introduce a directed acyclic graph (DAG) of size $K \times T \times L$ such as the one of Fig. 2, in which every node represents a location at a given time instant and for a particular identity group. Edges between nodes represent admissible motion between locations. Since groups cannot exchange their identity, there are no edges linking groups, that is, no vertical edges in Fig. 2. The resulting graph is made of disconnected layers, one per identity group.

Let $\mathcal{N}(k) \subset \{1, \dots, K\}$ be the neighborhood of k, that is, the locations that can be reached from k in one time instant. There is an edge $e^l_{i,j}(t)$ from node i to node j if and only if $j \in \mathcal{N}(i)$. We associate to every node of the graph a variable $m^l_k(t)$ standing for the number of people of group l present on location k at time t. Similarly, a variable $f^l_{i,j}(t)$ corresponds to every edge $e^l_{i,j}(t)$, and encodes the number of people of group l moving from node i to j at time t.

We now define a set of constraints to ensure that every flow through the graph is physically possible. First, we enforce flow continuity by making sure that the sum of flows arriving at one node at time t is equal to the sum of flows leaving the same location at time $t+1$

$$\forall t, j, l \quad \underbrace{\sum_{i:j\in\mathcal{N}(i)} f^l_{i,j}(t)}_{\text{Arriving at } j} = m^l_j(t) = \underbrace{\sum_{k\in\mathcal{N}(j)} f^l_{j,k}(t+1)}_{\text{Leaving from } j} . \quad (1)$$

Second, our grid resolution is sufficiently fine, for a location not be occupied by more than one person, hence

$$\forall t, k, \quad \sum_{j\in\mathcal{N}(k)} \sum_{l=1}^{L} f^l_{k,j}(t) \leq 1 . \quad (2)$$

Third, the flows have to be positive and we have

$$\forall k, j, t, \quad f^l_{k,j}(t) \geq 0 . \quad (3)$$

In case we have a precise knowledge about the number of people we track, we can use an optional constraint to ensure that no more than the allowed number of people is present in each group

$$\forall t, l \sum_{k=1}^{K} m^l_k(t) \leq N_l . \quad (4)$$

Our model as described so far can only handle a fixed number of people. In practice, however, this number is likely to vary over time. We therefore introduce a source and a sink nodes, v_{source} and v_{sink}. The source node is connected to every node from the first frame and the sink to every node from the last frame, as shown in Fig. 2. Additionally, both nodes are connected to all the locations susceptible to act as entry or exit points, throughout the whole sequence. This last part is not illustrated in Fig. 2 to avoid overloading the graph. The source and sink nodes are also subject to a constraint that enforces all the flows starting in v_{source} to end in v_{sink}

$$\sum_{j\in\mathcal{N}(v_{\text{source}})} f_{v_{\text{source}},j} = \sum_{k:v_{\text{sink}}\in\mathcal{N}(k)} f_{k,v_{\text{sink}}} . \quad (5)$$

3.2. Linear Program

Let us now assume that we have access to a person detector that estimates the probability of presence of someone at every position k

$$\rho_k(t) = \hat{P}(X_k(t) = 1 \,|\, \mathbf{I}) , \quad (6)$$

where $X_k(t)$ is a random variable standing for the true occupancy of location k at time t, and $\mathbf{I}$ represents the input images. Let us furthermore assume that we can compute an appearance model and that we use it to estimate

$$\varphi^l_k(t) = \hat{P}(Q_k(t) = l \,|\mathbf{I}, X_k(t) = 1) , \quad (7)$$

the probability that the identity of a person occupying location k at time t is l, given that the location is indeed occupied. Here, $Q_k(t)$ is a random variable standing for the true identity group of a person in location k at time t. Let there be L identity groups, hence $Q_k(t) \in \{1, \dots, L\}$. The appearance model can rely on various cues, such as color

similarity or shirt numbers of sports players. In Section 4.2, we describe in details the ones we use for different datasets.

Since we are seeking a set of physically possible trajectories that best explain the observed image evidence, we look for

$$\mathbf{m}^* = \arg\max_{\mathbf{m}\in\mathfrak{F}} P(\mathbf{X}=\mathbf{x}, \mathbf{Q}=\mathbf{q} \,|\, \mathbf{I}) \ , \tag{8}$$

where $\mathbf{m}$ is a set of occupancy maps and $\mathfrak{F}$ is the space of occupancy maps satisfying constraints from Eqs. 1 to 5.

As shown in the appendix supplied as supplementary material, Eq. 8 can be expressed as a function of $\rho_i(t)$ and $\varphi_i^l(t)$ as

$$\mathbf{m}^* = \arg\max_{\mathbf{m}\in\mathfrak{F}} \sum_{t,k,l} m_k^l(t) \log\left(\frac{\rho_k(t)\varphi_k^l(t)L}{1-\rho_k(t)}\right) . \tag{9}$$

Note that when no appearance information is available, we set $\forall l, \varphi_k^l(t) = \frac{1}{L}$ and the appearance term cancels the L coefficient in the objective function. In this case, this simplifies to the original one of [2]. This property makes it possible to process sparse appearance information, such as shirt numbers that can be read only once in a while. The spatial extent of trajectories is mostly based on the occupancy information, while the sparse appearance places a trajectory in the correct identity group and avoids switches at intersections.

Maximizing the criterion of Eq. 9 under the constraints of Eqs. 1 to 5 can be formulated as an Integer Program, which is optimized with respect to the flows $f_{i,j}^l(t)$

$$\begin{aligned}
&\text{maximize} \sum_{t,i,l} \log\left(\frac{\rho_i(t)\varphi_i^l(t)L}{1-\rho_i(t)}\right) \sum_{j\in\mathcal{N}(i)} f_{i,j}^l(t) \\
&\text{subject to } \forall t,i, \sum_{j\in\mathcal{N}(i)} \sum_{l=1}^{L} f_{i,j}^l(t) \le 1 \\
&\qquad \forall t,l,i, \sum_{j\in\mathcal{N}(i)} f_{i,j}^l(t) - \sum_{k:i\in\mathcal{N}(k)} f_{k,i}^l(t-1) \le 0 \\
&\qquad \sum_{j\in\mathcal{N}(v_{\text{source}})} f_{v_{\text{source}},j} - \sum_{k:v_{\text{sink}}\in\mathcal{N}(k)} f_{k,v_{sink}} \le 0 \\
&\qquad \forall t,l, \sum_{i=1}^{K} \sum_{j\in\mathcal{N}(i)} f_{i,j}^l(t) \le N_l \\
&\qquad \forall t,l,i,j, \ f_{i,j}^l(t) \ge 0 \ .
\end{aligned} \tag{10}$$

3.3. Optimization

The large number of variables and constraints of our formulation results in too large a problem to be directly handled by regular solvers for real-life cases such as those presented in Section 4. However, a simple way to bring the computational complexity down is to remove unnecessary nodes from the graph.

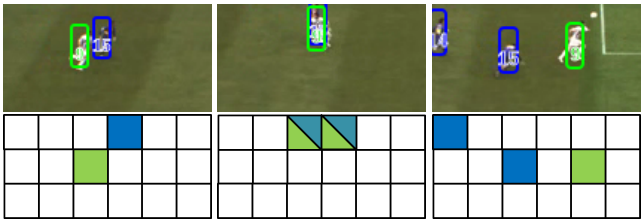


Figure 3. The LP solver might produce non-integer values when two or more people intersect. In this example, our tracking algorithm assigns non-integer values for the identities of the two soccer players at two adjacent locations. After the intersection, the algorithm recovers and assigns again integer value to each identity.

To this end, we first run the earlier K-Shortest Paths (KSP) algorithm [2], which is very efficient but ignores appearance and is therefore more prone to identity switches. This lets us eliminate all the grid cells in which nobody was found. We do this everywhere except at locations where trajectories come close to each other, which we define as being within three grid cells from each other. At these locations, we include all grid cells connecting both trajectories. The reason behind this is that KSP produces trajectories with very good spatial accuracy, except at places where people meet and separate. There, it may erroneously link bits of trajectories belonging to different individuals and ignore the grid cells through which the true trajectories pass. By adding the additional grid cells, we give our algorithm the degrees of freedom it requires to avoid such mistakes by using image evidence. In our experiments, the pruning reduces the number of variables and constraints by two to three orders of magnitude. We have performed a number of Monte Carlo simulations on small synthetic examples for which we can solve the problem without pruning and verified that it has almost no impact on the final accuracy.

Since Integer Programming solving is NP-complete, we relax our initial IP problem of Eq. 10 into a Linear Program, by allowing the variables to become real-valued. This results in a significant complexity reduction. The LP results however, are no longer guaranteed to be integral. In practice, real values might occur when two or more targets are moving so close to each other that appearance information is unable to disambiguate their respective identities, as shown in Fig. 3. These non-integer results can be interpreted as an uncertainty about identity assignment by our algorithm. This represents valuable information that can be dealt with accordingly if necessary. Note however that in our experiments those non-integer results occur only in rare occasions. In those cases, we currently round the non-integer results.

4. Experiments

We use multi-camera sequences acquired during soccer and basketball matches to validate our approach and compare it against the approach we extend [2], which completely ignores image appearance, and a modified version of it that takes frame-to-frame appearance into account, as described in [1]. Additionally, to compare our approach against other state-of-the-art ones, we test it on the PETS'09 benchmark dataset, which features pedestrians.

In the remainder of this section, we first describe these video sequences. We then discuss how we obtain image evidence and present our results [1].

4.1. Datasets

Team-sports players are hard to track reliably because they tend to converge towards the ball, often change their direction of travel abruptly, and wear the same uniforms when they belong to the same team. The only reliable way to identify them is to read the numbers on their shirts but, given the resolution of the images, this can only be done in relatively few frames. Furthermore, even though the color of the uniforms can be used to tell the teams apart, this information is hard to exploit at the most critical times, that is, when several players are bunched together.

Therefore, team sports sequences are challenging and we tested our approach on both basketball and soccer sequences, along with a standard pedestrian benchmarking dataset, which we describe in more detail below.

Basketball We acquired a 4,000-frame sequence at the 2010 FIBA World Championship for Women , using 8 cameras – 4 wide-angle ones, 2 looking from above, and 2 providing close-ups – filming at 25 fps. There are 14 people, 4 referees and coaches, and two teams of 5 players. For this dataset, we run two experiments: In the first one, we use only color as appearance information, and the identity groups consist thus of two teams and referees. In the second, we use number reading in addition to shirt colors, which allows to handle 11 groups - one per player and one group for the three referees and coach.

Soccer We use the publicly available ISSIA dataset [6]. It is made of 3,000 frames filmed by six cameras at a soccer match. There are 25 people, 3 referees and two teams of 11 players, including the goal keepers whose uniform is different from the one of their teammates. Due to the dataset resolution, the shirt numbers are unreadable. Hence, the appearance is based on shirt colors only. We use 5 identity groups that we denote as *referees*, *team 1*, *team 2*, *goal keeper 1* and *goal keeper 2*.

[1]For the supplementary material and videos, please visit: http://cvlab.epfl.ch/research/body/surv/

Pedestrians We use the publicly available PETS'09[2] dataset, for which the performance of other algorithms has been published [7]. More specifically, we tested our method on the 800-frame sequence `S2/L1`, which is filmed by 7 cameras at 7 fps, and features 10 people. In this sequence, the density of people is lower than in the two sport datasets but most of the pedestrians wear similar dark clothes, which makes appearance-based identification very challenging. We therefore used only two appearance groups, one for people wearing dark clothes and the other for those wearing reddish ones.

4.2. Implementation Details

Our system is implemented in C++ using standard libraries. To produce the Probability Occupancy Maps (POMs) we need as input, we use the publicly available POM software package[3]. It implements an algorithm that estimates ground plane occupancy from the binary output of a background subtraction algorithm in multiple images acquired simultaneously using calibrated cameras [8]. The LP problems were formulated and optimized using the MOSEK[4] solver. The average running time of our method is 4 seconds per frame on a 3GHz PC using a single core, which makes it practical to process whole batches at once.

We exploit two distinct sources of image information, the color of the uniforms and the numbers on the players shirts. This is done as follows.

Color Similarity Since our sequences feature groups – players of the same team, referees – whose appearance is similar, we manually select a few POM-generated bounding boxes corresponding to members of that group, convert the foreground pixels within each box to the CIE-LAB color space, and use them to populate a $20 \times 20 \times 20$ color histograms. We repeat this process independently for each camera because they are not color calibrated.

Extracting color information from closely spaced people is unreliable because it is often difficult to correctly segment individuals. Thus, at run time, for each camera and at each time frame, we first compute an occlusion map based on the raw probability occupancy map: If a specific location is occluded with high probability in a given camera view, we do not use it to compute color similarity for this location. Within a detection bounding box, we use the background subtraction result to segment the person. The segmented pixels are inserted into a color histogram, in the same way as for template generation. Finally, the similarity between this observed color histogram O_{colors} and the templates T_{colors} is computed using the Kullback-Leibler divergence. For each location, the final classification score

[2]PETS 2009: http://www.cvg.rdg.ac.uk/PETS2009

[3]POM: http://cvlab.epfl.ch/software/pom

[4]Mosek Optimization tool: http://www.mosek.com/

(i) (ii) (iii) (iv)

Figure 4. Examples of color projections: (i) color image , (ii) gray-scaled image, (iii) color projection using the two colors of the green team, (iv) color projection using the two colors of the white team. These projections allow us better recognitions.

is the average over the maximum matching scores from the non-occluded views v. We normalize this term in order to get a probability between 0 and 1.

$$\varphi_i^{t,l} \propto \frac{\sum_v \exp(-\mathrm{KL}(T_{\text{colors}}, O_{\text{colors}}))}{|v|} \quad . \tag{11}$$

If no appearance cue is available, due to occlusions for example, $\varphi_i^{t,l}$ is set to $\frac{1}{L}$.

Reading the Numbers The numbers on the back of sports players are unique identifiers, and can be used to unambiguously recognize them. Within a team, the printed numbers usually share a unique color, which is well separated from the shirt color. Here we use this observation to develop a specific image binarization that improves number recognition. For every team, the shirt color c_s and number color c_n are obtained by clustering a shirt color patch into two clusters. Then, for each pixel we measure the distance between its color c_p and these two colors: $d_s = ||c_s - c_p||, d_n = ||c_n - c_p||$. The converted gray-level pixel is defined as $255\frac{d_n}{d_s+d_n}$, which produces a white number on a black shirt. An illustration of this projection method is shown in Fig. 4. Finally, we binarize those images.

As for group classification, we manually extract a template for every player beforehand. At run time, applying number recognition at every position of an image would be much too expensive. Instead, we rely on people detection to select candidate positions for number reading. For each candidate position, we trim the upper $1/5$ part and the lower $1/5$ part of the bounding box, which roughly correspond to the head of the player and his legs respectively. We then search for number candidates inside the reduced bounding box, by using XOR operation between the templates and observation patches with the same size.

We select the observation patch that gives us the maximum normalized sum of pixel-wise XOR between the template and the observation and write

$$\varphi_i^{t,l} \propto \frac{T_{\text{numbers}} \oplus O_{\text{numbers}}}{|T_{\text{numbers}}|} \quad . \tag{12}$$

Since numbers cannot be read often, we favor highly confident detections. Therefore, we only keep scores that are higher than a threshold, 0.8 in our case. In other cases, we set $\varphi_i^{t,l}$ to a neutral value of $\frac{1}{L}$.

4.3. Baseline

As a baseline, we use the publicly available[5] KSP tracker [2], that ignores appearance. Nevertheless, it has been shown to outperform many state-of-the-art methods on the PETS'09 dataset [7]. In addition, we use a modified version of the KSP that includes appearance information from frame to frame. This method we will refer to as *C-KSP* only differs from the original algorithm in the cost of the edges. It includes an appearance term $\zeta_{i,j}^t$ in addition to the detection term

$$c(e_{i,j}(t)) = -\log(\frac{\rho_i^t \zeta_{i,j}^t L}{1-\rho_i^t}) \quad . \tag{13}$$

The appearance term $\zeta_{i,j}^t$ is generated the same way as $\varphi_i^{t,l}$, between color histograms from two locations in successive frames, similarly to what is done in [1].

4.4. Evaluation Metrics

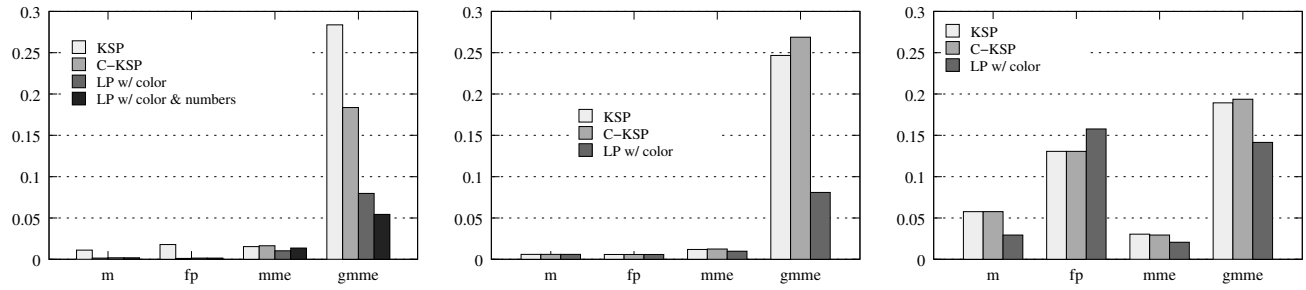
A standard metric for evaluating object trackers is the Multiple Object Tracking Accuracy (MOTA) [3], defined as

$$\text{MOTA} = 1 - \frac{\sum_t (c_m(m_t) + c_f(fp_t) + c_s(mme_t))}{\sum_t g_t} \quad ,$$

where g_t is the number of ground truth detections, m_t the number of miss-detections, fp_t the false positive count and mme_t the number of *instantaneous* identity switches. According to [12], the weighting functions are set to $c_m = c_f = 1$, and $c_s = \log_{10}$. While providing a reliable performance measure for generic tracking systems, this metric is not appropriate to evaluate applications for which identity preserving is crucial. Its mme term penalizes only instantaneous identity switches, that is the frame at which two trajectories are switched, but does not account for the proportion of a trajectory that is correctly labeled over a whole sequence.

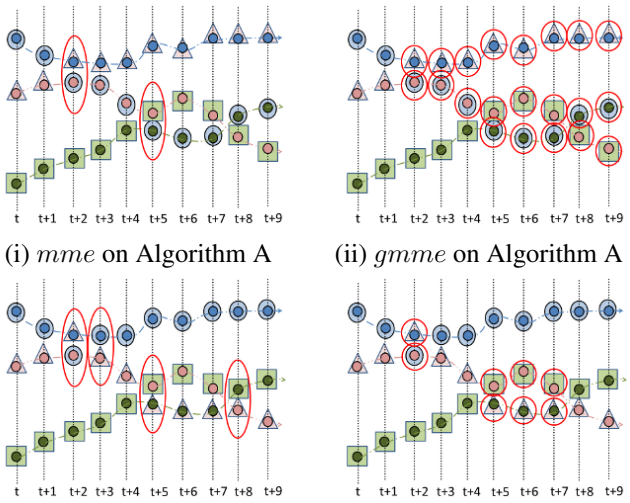
Therefore, we introduce a new term $gmme$ for measuring the proportion of identity switches in a global manner. For every detection at every frame, the $gmme$ term is incremented if the detection label does not correspond to the ground truth identity. Thus, a trajectory with an identity switch in the middle will be counted wrong for half of its length, instead of just once for the mme, as explained on Fig. 5. We could generate a new metric, by replacing mme by $gmme$ in MOTA, but for the sake of clarity, we will show results on each of the components m, fp, mme and $gmme$, in next section.

[5] KSP: http://cvlab.epfl.ch/software/ksp



(a) Basketball (b) Soccer (c) Pedestrians

Figure 6. Performance comparison of our method (LP) against the baseline (KSP and C-KSP). We plot separately each component of MOTA: the miss-detections and false positive rates m and fp, the rate of instantaneous identity switches mme, and the rate of global identity mismatch $gmme$. While MOTA components (m, fp, mme) are similar among the three algorithms, our method is much better at preserving identities, as reflected by the low $gmme$ rates. Note that, for these scores, lower is better.



(i) mme on Algorithm A (ii) $gmme$ on Algorithm A

(iii) mme on Algorithm B (iv) $gmme$ on Algorithm B

Figure 5. Illustration of the difference between identity mismatch score mme and global identity mismatch score $gmme$. We apply the two scores on two synthetic tracking results A and B. The mismatches are circled in red. As can be seen, algorithm B manages to recover from its tracking mistakes. However, its mme score is worse than the one of Algorithm A. Our proposed $gmme$ score favors algorithms that preserve identities.

4.5. Results

We ran our algorithm and the baseline, KSP and C-KSP, on our three datasets. Evaluated with the MOTA metric, the three algorithms all exhibit excellent performances, as shown in Fig. 8. Our algorithm is always either as good or better than KSP, which has itself been shown to outperform state-of-the-art methods on the PETS'09 sequence [7].

To better understand the performance of the different algorithms, we then compute the three individual components of MOTA m, fp, mme, as well as the new global identity mismatch score $gmme$. The results are plotted on Fig. 6 for the three datasets. Those results show rather similar missed detection and false positive rates for all methods. Also, the mme term is uniformly low, which explains the similarity of the MOTA results. By contrast, the more accurate $gmme$ metric clearly indicates the performance difference between algorithms in terms of identity assignment: Our method is shown to preserve the identities much better than the two baselines. What is more, the addition of a second appearance cue – numbers on the back of players – is shown to further decrease the already low amount of identity mismatch, on the basketball dataset (Fig. 6(a)). The performance of the C-KSP is more erratic: It improves over the KSP on the basketball data set, but is slightly worse on the two others. This shows that a frame-by-frame appearance constraint is not enough to preserve identities over a long period of time. The improvement of our approach over the two baseline methods is less important on the PETS'09 pedestrian dataset than on the sport sequences. The reason is that people in this sequence are wearing dark clothes of a relatively uniform color.

In general, when the appearance information is less discriminant, failure cases such as the one illustrated by Fig. 7 are more likely to happen. In case no appearance information is available, the optimization is based purely on the geometrical constraints, similarly to the KSP algorithm. Note that for the pedestrian dataset, a motion model such as the one from [1] would probably reduce the number of identity switches. However, it would not be applicable to the sport datasets, where players' movements are way too erratic.

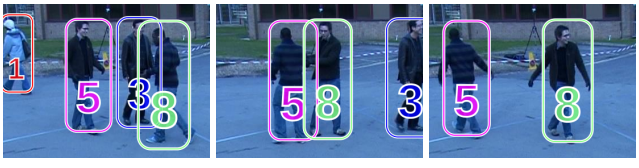


Figure 7. Failure case: Despite the global appearance model, individuals 5 and 8 are switched because of similarly colored clothes.

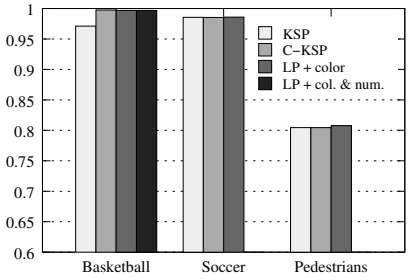


Figure 8. Evaluation of our method (LP) against the baseline (KSP and C-KSP) using the MOTA metric. While excellent, the scores are almost the same for all methods. This is because MOTA only considers instantaneous identity switches, and weights them by $\log_{10}$. Note that higher values are better and the maximum is 1. More detailed results are presented in Fig. 6.

5. Conclusion

In this paper, we introduced a global optimization framework for multi-people tracking that takes image-appearance cues into account, even if they are only available at distance time intervals. As a result, it does better at preserving identity over very long sequences than previous approaches. Furthermore, it depends on a comparatively small number of parameters such as the size of the grid it works on and the maximum number of separate identities to be expected.

Future work will focus on automatically estimating the number of appearance groups by clustering the detections and exploiting the fact that our algorithm occasionally returns non-integer probabilities to invoke more sophisticated domain-knowledge only at critical junctures.

References

[1] A. Andriyenko and K. Schindler. Globally Optimal Multi-Target Tracking on a Hexagonal Lattice. In *ECCV*, 2010.

[2] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua. Multiple Object Tracking Using K-Shortest Paths Optimization. *PAMI*, 2011.

[3] K. Bernardin and R. Stiefelhagen. Evaluating Multiple Object Tracking Performance: the Clear Mot Metrics. *EURASIP Journal on Image and Video Processing*, 2008.

[4] J. Black, T. Ellis, and P. Rosin. Multi-View Image Surveillance and Tracking. In *IEEE Workshop on Motion and Video Computing*, 2002.

[5] S. Blackman. *Multiple-Target Tracking With Radar Applications*. Artech House, 1986.

[6] T. D'Orazio, M. Leo, N. Mosca, P. Spagnolo, and P. L. Mazzeo. A Semi-Automatic System for Ground Truth Generation of Soccer Video Sequences. In *AVSS*, 2009.

[7] A. Ellis, A. Shahrokni, and J. Ferryman. Pets 2009 and Winter Pets 2009 Results, a Combined Evaluation. In *PETS*,2009.

[8] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-Camera People Tracking With a Probabilistic Occupancy Map. *PAMI*, 2008.

[9] J. Giebel, D. Gavrila, and C. Schnorr. A Bayesian Framework for Multi-Cue 3D Object Tracking. In *ECCV*, 2004.

[10] S. Iwase and H. Saito. Parallel Tracking of All Soccer Players by Integrating Detected Positions in Multiple View Images. In *ICPR*, 2004.

[11] H. Jiang, S. Fels, and J. Little. A Linear Programming Approach for Multiple Object Tracking. In *CVPR*, 2007.

[12] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, M. Boonstra, V. Korzhova, and J. Zhang. Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics, and Protocol. *PAMI*, 2009.

[13] Z. Khan, T. Balch, and F. Dellaert. MCMC-Based Particle Filtering for Tracking a Variable Number of Interacting Targets. *PAMI*, 2005.

[14] D. R. Magee. Tracking Multiple Vehicles Using Foreground, Background and Motion Models. *Image and Vision Computing*, 2004.

[15] E. Maggio, M. Taj, and A. Cavallaro. Efficient Multi-Target Visual Tracking Using Random Finite Sets. *IEEE Transactions On Circuits And Systems For Video Technology*, 2008.

[16] T. Mauthner, M. Donoser, and H. Bischof. Robust Tracking of Spatial Related Components. In *ICPR*, 2008.

[17] T. Misu, A. Matsui, S. Clippingdale, M. Fujii, and N. Yagi. Probabilistic Integration of Tracking and Recognition of Soccer Players. In *Advances in Multimedia Modeling*, 2009.

[18] A. Mittal and L. Davis. M2Tracker: a Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene. *IJCV*, 2003.

[19] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe. A Boosted Particle Filter: Multitarget Detection and Tracking. In *ECCV*, 2004.

[20] A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and H. Wensheng. Multi-Object Tracking through Simultaneous Long Occlusions and Split-Merge Conditions. In *CVPR*, 2006.

[21] H. Pirsiavash, D. Ramanan, and C. Fowlkes. Globally-Optimal Greedy Algorithms for Tracking a Variable Number of Objects. In *CVPR*, 2011.

[22] K. Smith, D. Gatica-Perez, and J.-M. Odobez. Using Particles to Track Varying Numbers of Interacting People. In *CVPR*, 2005.

[23] P. P. A. Storms and F. C. R. Spieksma. An LP-Based Algorithm for the Data Association Problem in Multitarget Tracking. *Computers & Operations Research*, 2003.

[24] J. W. Suurballe. Disjoint Paths in a Network. *Networks*, 1974.

[25] J. Wolf, A. Viterbi, and G. Dixon. Finding the Best Set of K Paths through a Trellis With Application to Multitarget Tracking. *IEEE Transactions on Aerospace and Electronic Systems*,1989.

[26] M. Xu, J. Orwell, and G. Jones. Tracking Football Players With Multiple Cameras. In *ICIP*, 2004.

[27] C. Yang, R. Duraiswami, and L. Davis. Fast Multiple Object Tracking Via a Hierarchical Particle Filter. In *ICCV*, 2005.

[28] L. Zhang, Y. Li, and R. Nevatia. Global Data Association for Multi-Object Tracking Using Network Flows. In *CVPR*, 2008.