Tracking and Labelling of Interacting Multiple Targets

Josephine Sullivan and Stefan Carlsson

Royal Institute of Technology, Stockholm, Sweden
{sullivan, stefanc}@nada.kth.se

Abstract. Successful multi-target tracking requires solving two problems - *localize* the targets and *label* their identity. An isolated target's identity can be unambiguously preserved from one frame to the next. However, for long sequences of many moving targets, like a football game, grouping scenarios will occur in which identity labellings cannot be maintained reliably by using continuity of motion or appearance. This paper describes how to match targets' identities despite these interactions.

Trajectories of when a target is isolated are found. These trajectories end when targets interact and their labellings cannot be maintained. The interactions (merges and splits) of these trajectories form a graph structure. Appropriate feature vectors summarizing particular qualities of each trajectory are extracted. A clustering procedure based on these feature vectors allows the identities of temporally separated trajectories to be matched. Results are shown from a football match captured by a wide screen system giving a full stationary view of the pitch.

1 Introduction

This paper addresses the problem of the surveillance and tracking of multiple persons over a wide area. Typical scenarios involve following pedestrians in traffic or other crowded environments such as airports and shopping malls. There is also the more specialized problem of tracking players in team sports, such as football or ice-hockey, which we specifically explore. The first challenge is defining a multiple camera set-up that ensures all the target objects are visible, at a required resolution, at all times. Such a camera set-up is shown in figure 2.

Over the last years, many algorithms and results have been presented [1, 2] with regard to the problem of multiple object tracking. Prevalent are algorithms based on kalman filtering [3, 4] and advanced techniques of particle filtering [5, 6, 7, 8]. These works demonstrate that the tracking of individual players is no problem as long as they are isolated. Situations of congestion and confusion due to multiple players occluding each other are generally resolved by exploiting continuity of motion, appearance and relative depth. However, these properties cannot be used to reliably solve all congested situations, see fig. 1. It is clear even extremely sophisticated trackers will eventually lose the identity of a track due to the complex scenarios that arise during a soccer game lasting 90 minutes. A

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part III, LNCS 3953, pp. 619–632, 2006.

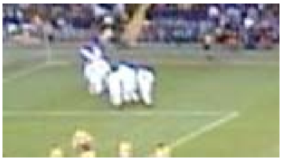


Fig. 1. All the players from one team have congregated into a small area to celebrate a goal. In this situation the players' identities cannot be maintained by using continuity of motion, appearance or relative depth, irrespective of the camera viewpoint.

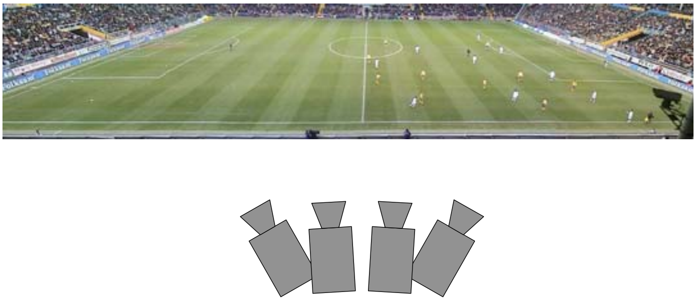


Fig. 2. Four cameras are on one side of the field. Each one looks at a portion of the pitch and the images obtained are stitched together using the inter camera homographies.

system for the automatic tracking of players over a whole game must, therefore, address the problem of automatic track label initialization and re-initialization.

With these observations in mind, consider the strategy put forward in this paper. Periods and trajectories when players are isolated are identified, which we term *player tracks.* The merge and splits between the *player tracks* are recorded to form a giant graph summarizing the game. A feature vector for each track, encoding a player's relative spatial position wrt his team-mates, is defined. The identity of the *player tracks* are then linked through analysis of the recorded merge and splits and clustering of the feature vectors associated with each track.

Paper Overview. The paper is organized as follows. Section 2 is devoted to explaining the imaging set-up and image processing performed to get an initial estimate of the player/target positions. The machinery used to temporally analyze these results is then introduced in addition to describing how a graph summarizing the interaction of the targets is obtained. Section 4 is concerned with resolving some of the split and merge situations that occur using the properties of continuity of motion, appearance and relative depth ordering. Finally a clustering procedure is described that allows the identity of temporally separated player trajectories to be linked. The paper ends with a discussion on the merits of the ideas put forward here and future avenues of research. Throughout results are displayed from analyzing 10 minutes of an international football match.

2 Video Capture and Processing

One of the innovations of this paper is the use of wide-screen video in our tracking/labelling algorithm. This video allows a wide field of view to be monitored at each time instant at a high resolution. This makes it possible to track simultaneously a large group of targets spread over a large area. This wide screen video is obtained by mounting several cameras (in our case 4) on a tripod with each camera directed towards a portion of the area of interest. As the optical centres of the cameras are aligned a homography relates the images between each camera. This imaging set-up allows us a full, stationary view of the area-of-interest, an ideal environment to perform background subtraction given a background image model. The next subsection describes how to find this background model.

2.1 Background Modelling

A probabilistic model for the image gradient of each pixel in the background is obtained. Note that the image gradient is learnt as opposed to the rgb values, as often a football team's uniform contains white, the colour of the pitch markings. Also using the image gradient increases robustness to changes in illumination.

Let $\mathbf{g}_x^t$ denote the image gradient at pixel x in frame t. Each background pixel's image gradient, $\mathbf{g}_x^b$, is modelled by a bivariate normal distribution with mean μ_x and diagonal covariance matrix Σ_x. The parameters of this distribution are learnt in the manner of Stauffer and Grimson [9]. Except we consider it as a batch process and learn the background for a time interval. The initial learning algorithm produces, for each pixel x, a mixture of Gaussians distribution to describe the values of $\mathbf{g}_x$ over a set of training images.

$$\mathbf{g}_x \sim \sum_{i=1}^{3} \beta_x^i \ \mathbf{N}_2(\mu_x^i, \Sigma_x^i) \tag{1}$$

with each $0 \leq \beta_x^i \leq 1$ and the β_x^i's summing to one. It is assumed that at a given location, x, the background is most commonly visible. To ensure this assumption

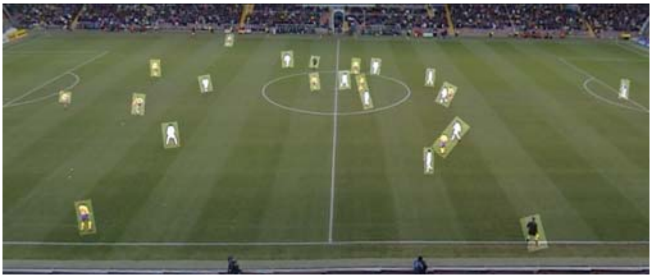


Fig. 3. The set of ellipses, $\mathcal{E}_t$, the highlighted regions, found by background subtraction

is true, as players frequently stand still for relatively lengthy periods of time, for a given time window of length T only every nth frame is used as input. Then set

$$(\mu_x, \Sigma_x) = (\mu_x^i, \Sigma_x^i) \text{ s.t. } \beta_x^i \geq \beta_x^j \text{ for } j = 1, 2, 3. \tag{2}$$

For a frame in the time window a pixel, x, is considered a foreground pixel if:

$$(\mathbf{g}_x^t - \mu_x)^T \Sigma_x^{-1} (\mathbf{g}_x^t - \mu_x) \geq \chi^2_{3,\alpha}, \quad 0 < \alpha < 1 \tag{3}$$

Let $\mathcal{F}_t$ ($\mathcal{B}_t$) be the set of foreground(background) pixels found at time t. Once $\mathcal{F}_t$ has been calculated, connected components are identified. A set $\mathcal{E}_t = \{E_t^i\}_{i=1}^{n_t}$ of ellipses is used to represent these components, see figure 3. We assume that each ellipse corresponds to at least one whole player. The raw connected components are processed by deleting small connected components or joining them to neighbouring larger ones to ensure this assumption is for the most part upheld.

3 Constructing the Target Interaction Graph

This section is devoted to the temporal analysis of the $\mathcal{E}_t$'s - spotting the merging and splitting of ellipses from one frame to the next. Detection of these splits and merges allows for the identification of ellipses corresponding to individual and multiple targets and then to the trajectories of individual players.

3.1 Finding Player Tracks

The first aim is to put the ellipse in $\mathcal{E}_t$ and $\mathcal{E}_{t+1}$ in correspondence. This requires the definition of a relation, $\sim$, between ellipses in $\mathcal{E}_t$ and $\mathcal{E}_{t+1}$. Let ellipses E_1 and E_2 be an exact match if their size and orientation are sufficiently similar and the displacement between their centres is sufficiently small. If $\exists E_{t+1}^j \in \mathcal{E}_{t+1}$ s.t. E_t^i and E_{t+1}^j are an exact match then $E_t^i \sim E_{t+1}^j$. If no such exact match exists for E_t^i in $\mathcal{E}_{t+1}$ then $E_t^i \sim E_{t+1}^j$ if

$$\text{Area}(E_t^i \cap E_{t+1}^j) > 0 \quad \& \quad E_{t+1}^j \text{ has no exact match in } \mathcal{E}_t. \tag{4}$$

Define a mapping F_t, matching each ellipse in $\mathcal{E}_t$ to its related ellipses in $\mathcal{E}_{t+1}$

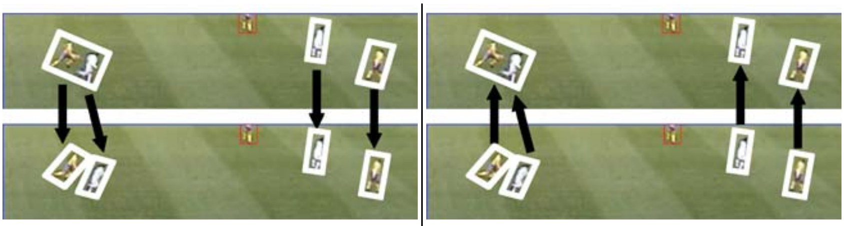
$$F_t : \mathcal{I}_t \rightarrow \{a : a \subset \mathcal{I}_{t+1}\} \quad \text{s.t.} \quad j \in F_t(i) \Rightarrow E_t^i \sim E_{t+1}^j \tag{5}$$

where $\mathcal{I}_t = \{1, \cdots, n_t\}$ is the set of indices of the ellipses in $\mathcal{E}_t$ (see figure 4). A mapping B_t is then defined relating each ellipse in $\mathcal{E}_t$ to a subset of $\mathcal{E}_{t+1}$,

$$B_t : \mathcal{I}_{t+1} \rightarrow \{a : a \subset \mathcal{I}_t\} \quad \text{s.t.} \quad k \in B_t(i) \Rightarrow i \in F_t(k)\,. \tag{6}$$

With F_t and B_t it is easy to define events that can happen or have happened at each ellipse at each frame given the cardinality of $F_t(i)$ and $B_t(i)$:

Signal	Event	Signal	Event
$\lvert F_t(i)\rvert > 1$	*split*	$\lvert B_t(j)\rvert > 1$	*merge*
$\lvert F_t(i)\rvert = 0$	*disappear*	$\lvert B_t(j)\rvert = 0$	*appear*
$\lvert F_t(i)\rvert = \lvert B_t(F_t(i))\rvert = 1$	*stable*		



(a) F_t

(b) B_t

Fig. 4. The correspondences between the ellipses in one frame and those in the next

A maximal sequence of *stable* events sandwiched between non-*stable* events is termed a *track*. Formally, it is a temporal sequence, starting at t of n ellipses with indices $\mathbf{k}$

$$\mathcal{T}(t, n, \mathbf{k}) = \{E_t^{k_0}, E_{t+1}^{k_1}, \cdots, E_{t+n-1}^{k_{n-1}}\} \tag{7}$$

that unambiguously match to one another s.t. for $i = 0, \cdots, n-2$

$$|F_{t+i}(k_i)| = 1, \quad F_{t+i}(k_i) = k_{i+1}, \quad |B_{t+i}(k_{i+1})| = 1 \tag{8}$$

and a non-*stable* event occurs to go from $\mathcal{E}_{t-1}$ to $E_t^{k_0}$ and from $E_{t+n-1}^{k_{n-1}}$ to $\mathcal{E}_{t+n}$. During a track there is no change in the identity or number of the subjects represented by the ellipses concerned. It is clear, therefore, that if either of the following sequence of events occurs

$$\textit{track} \rightarrow \textit{split} \quad \text{or} \quad \textit{merge} \rightarrow \textit{track} \tag{9}$$

the involved *track* corresponds to multiple players. However, when the involved track appears in the sequence

$$\{\textit{split, appear}\} \rightarrow \textit{track} \rightarrow \{\textit{merge, disappear}\} \tag{10}$$

it may correspond to exactly one player. If the track has long enough duration and the size of the ellipses during the track is on average not too big then the track is considered to be a *player track*. Each ellipse in a player track corresponds to exactly one player and the identity of this player remains fixed over the track.

The application of this analysis results in a partition of all ellipses, found from the background subtraction process, into *player tracks* or *multiple player tracks* and the interactions of these tracks through merges and splits.

Modelling Player Appearance. Frequently in multi-target applications the appearance of some or all of the targets will be distinctive. In a football game there are two teams and officials with distinct uniforms. With a pdf for the rgb values for each category it is possible to distinguish between them. These distributions, $p_{A,B,C}$, are learnt from a few labelled training examples. The likelihood that ellipse E_t^j is team A is defined, naively, as:

$$p(E_t^j|A) = \prod_{x \in E_j^t} p_A(\mathbf{I}_t^x) \tag{11}$$

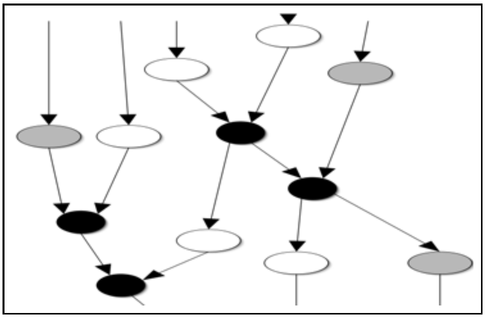


Fig. 5. The player interaction graph. White/gray nodes correspond to team A/B player tracks and black nodes to multiple player tracks. Edges indicate when tracks interact. Shown is a small section of the ∼ 5000 node graph describing the 10 minutes analyzed.

where $\mathbf{I}_t^x$ represents the rgb values of pixel x at time t. The ellipse is classified as the category with the maximal likelihood. Each ellipse in a player track, $\mathcal{T}(t, n, \mathbf{k})$, is classified accordingly. The label, λ, of the track is set to the category that occurs most frequently amongst its ellipses. All these ellipses are then set to the track label. As player tracks are quite long there is sufficient temporal evidence to compensate for less than perfect appearance models. Given the labelling of the tracks and their interactions through merging and splitting, the game can be summarized by a graph structure, see figure 5.

4 Linking Player Tracks

By examining the player interaction graph it is possible to isolate situations where n individual player tracks merge (potentially staggered over time) and then split into n individual player tracks (potentially staggered over time). These merge-split situations are resolved by finding the correct correspondence between the input and output tracks. This can often be done by exploiting the continuity of motion, appearance and/or relative depth ordering of the players involved.

4.1 Matching Input and Output Tracks

The set of input and output tracks with labels to be put in correspondence are:

$$\textbf{Input}\text{: } \{\mathcal{T}(t_{s_i}, n_{s_i}, \mathbf{k}^{s_i}), \lambda_{s_i}\}_{i=1}^n \quad \textbf{Output}\text{: } \{\mathcal{T}(t_{f_i}, n_{f_i}, \mathbf{k}^{f_i}), \lambda_{f_i}\}_{i=1}^n \tag{12}$$

For brevity we refer to $\mathcal{T}(t_{s_i}, n_{s_i}, \mathbf{k}^{s_i})$ as $\mathcal{T}_{s_i}$. We wish to find the assignment, M, of the inputs to the outputs. It is a bijective mapping $M : \{1, \cdots, n\} \rightarrow \{1, \cdots, n\}$ s.t. $M(i) = j$ implies that track $\mathcal{T}_{s_i}$ and $\mathcal{T}_{f_j}$ are the same player. Not all assignments are physically possible, thus M is a valid assignment iff all the input tracks and their matched output tracks have the same label and all the input tracks finish before their matched output tracks begin. Finding the correct assignment from the valid ones involves scoring each valid assignment and choosing the most plausible one. Our score is computed as follows.

For each valid assignment, M, we estimate the intermediate trajectories, $\mathcal{T}_{s_i \to f_{M(i)}}$'s between the matched player tracks $\mathcal{T}_{s_i}$ and $\mathcal{T}_{f_{M(i)}}$. There are numerous reasonable ways in which each $\mathcal{T}_{s_i \to f_{M(i)}}$ can be estimated. We return to this issue later in the section. For now we continue with the discussion of finding the correct valid assignment, assuming we have estimates for the $\mathcal{T}_{s_i \to f_{M(i)}}$'s. We define a score based on these estimated trajectories.

$$\mathrm{Sc}_M = \sum_{i=1}^{n} (\mathrm{Dist}(\mathcal{T}_{s_i \to f_{M(i)}}) + \alpha \mathrm{Pen}(\mathcal{T}_{s_i \to f_{M(i)}})) \tag{13}$$

where $\alpha > 0$ and has a large value. The first term $\mathrm{Dist}(\mathcal{T}_{s_i \to f_{M(i)}})$ is a measure of the distance traveled by the player during the hypothesized trajectory, measured using the estimated feet positions on the ground plane transformed via a homography to a rectified version of the pitch. The second term indicates whether the estimated trajectories are consistent with the image data:

$$\mathrm{Pen}(\mathcal{T}_i) = \begin{cases} 1 & \text{if } \mathcal{T}_i \text{ not consistent with relevant } \mathcal{F}_t\text{'s} \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

Due to space constraints we summarize in words our consistency measure. Its definition is based on deciding whether a sufficient number of a trajectory's ellipses intersect with a sufficient number of the foreground pixels. Once these scores have been calculated the valid assignment which does not incur the penalty α and whose intermediary trajectories cover the least distance is chosen.

4.2 Estimating Intermediary Tracks

We now focus on the task of estimating the intermediary trajectories $\{\mathcal{T}_{s_i \to f_{M(i)}}\}$. We exploit the properties of maintaining continuity of motion and relative depth ordering. At the first stage, we investigate if any of the intermediary tracks can

Data: A set of input and output player tracks as in eqn (12).

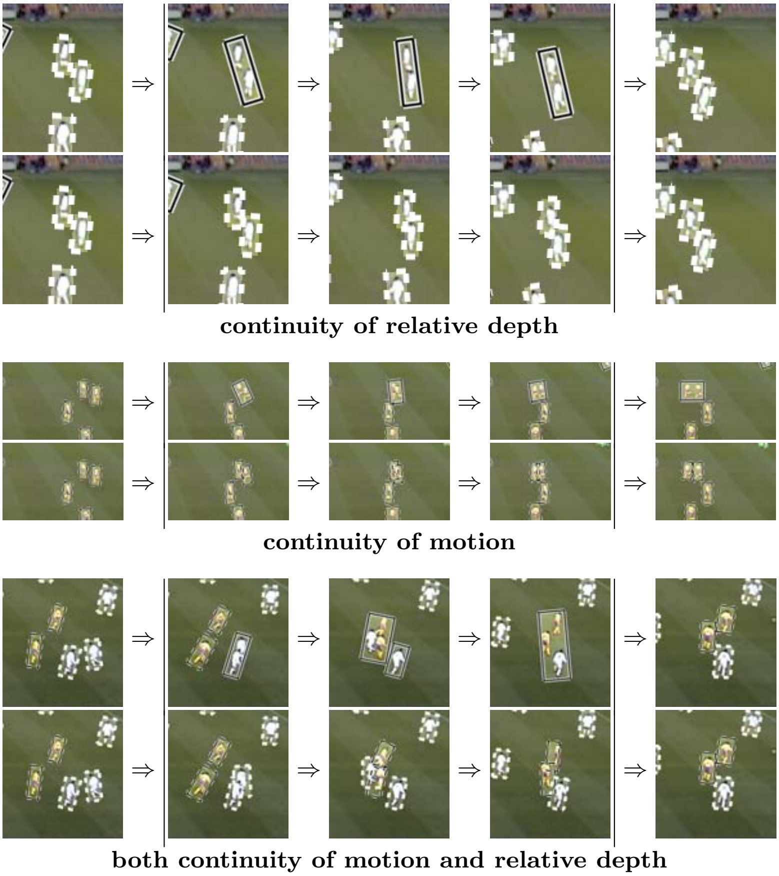
Algorithm:

1. Enumerate all the valid assignments $\{M_k\}_{k=1}^{K'}$.
2. For each M_k estimate the intermediary trajectories $\{\mathcal{T}_{s_i \to f_{M_k(i)}}\}$ based solely on linear interpolation. Score each assignment, eqn. (13), to obtain $\{\mathrm{Sc}_{M_k}\}$.
3. $k' = \arg\min_k \mathrm{Sc}_{M_k}$, if $\mathrm{Sc}_{M_{k'}} < \alpha$ set $M = M_{k'}$ and go to step 6.
4. If $\mathrm{Sc}_{M_{k'}} \geq \alpha$ repeat the process of finding the intermediary trajectories, but based on piecewise linear interpolation. Update the set of scores accordingly.
5. $k' = \arg\min_k \mathrm{Sc}_{M_k}$, if $\mathrm{Sc}_{M_{k'}} < \alpha$ set $M = M_{k'}$.
6. If M has been defined, for $i = 1, \cdots, n$ set

$$\mathcal{T}_{s_i} = \mathcal{T}_{s_i} \cup \mathcal{T}_{s_i \to f_{M(i)}} \cup \mathcal{T}_{f_{M(i)}}. \tag{15}$$

Update the interaction graph as the matched tracks have been concatenated.

Fig. 6. Algorithm for resolving merge-split scenarios in the player interaction graph



continuity of relative depth

continuity of motion

both continuity of motion and relative depth

Fig. 7. Examples of resolved merge and split situations. In each example the first row shows the split and merge scenario and the bottom row the found intermediary tracks.

be adequately described by a constant velocity motion model. To do this we linearly interpolate between the parameters of the last ellipse of $\mathcal{T}_{s_i}$ and the first ellipse of $\mathcal{T}_{f_{M(i)}}$. If there is sufficient image data evidence to support this trajectory, that is $\text{Pen}(\mathcal{T}_{s_i \rightarrow f_{M(i)}}) = 0$, it is considered as feasible. Let $\mathcal{Z}_M$ be the set of the indices of the input tracks whose trajectory to its matched output track cannot be modeled by a constant velocity trajectory.

The intermediate tracks for the elements of $\mathcal{Z}_M$ have to be estimated. It is at this stage we impose maintaining relative depth ordering amongst players from the same team. Every mth frame in the interval $[t_S, t_F]$, the temporal extent of the intermediate trajectories, is considered for analysis, with $t_k = t_S + km$. For each t_k let the subset $\mathcal{Z}_M^k \subset \mathcal{Z}_M$ be the set of tracks that exists at this time instant. Let E_j^F be the final ellipse in each trajectory $\mathcal{T}_{s_j}, j \in \mathcal{Z}_M$. Then define the region $\mathcal{R}_k(\{\mathbf{t}_j\})$ as the union of the ellipses $E_j^F, j \in \mathcal{Z}_M^k$ each displaced by $\mathbf{t}_j$. The aim at each t_k is to find the $\mathbf{t}_j$'s to maximize the intersection of $\mathcal{R}_k(\{\mathbf{t}_j\})$

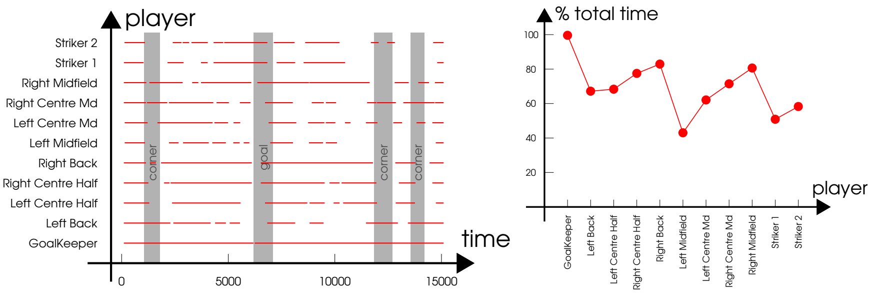


Fig. 8. The left graph shows the temporal extent of the player tracks (≥ 250 frames) for team A players during the 10 minutes of the game examined. Events causing congestion are marked by the gray strips. The right graph shows for each player the percentage of time it has been assigned to a player track. For the remaining frames the players are assigned to multiple player tracks or short player tracks.

with the foreground pixels and minimize its intersection with the background pixels or mathematically to maximize wrt the $\mathbf{t}_j$'s:

$$\alpha_1 \,\text{Area}(\mathcal{R}_k(\{\mathbf{t}_j\}) \cap \mathcal{F}_{t_k}) - (1 - \alpha_1)\,\text{Area}(\mathcal{R}_k(\{\mathbf{t}_j\}) \cap \mathcal{B}_{t_k}) \tag{16}$$

with $\alpha_1 > 0$, subject to the constraint that the depth ordering amongst players from the same team in $\mathcal{Z}_M$is maintained from t_S throughout the t_k's. Given the translations at each t_k the full trajectories $\{\mathcal{T}_{s_j \rightarrow f_{M(j)}}\}_{j \in \mathcal{Z}_M}$ are computed by interpolating between the displaced ellipses found at the fixed times.

We approximate this global optimization in a greedy manner. We first find the translation for the player closest to the camera by ensuring the displaced ellipse explains the relevant foreground pixels closest to the camera. Then similarly the translation for the player furthest away is found and then the inner players. It should be noted that we only analyze cases with $n \leq 5$, this generally implies that there are no more than 3 players from a single team are present. Figure 6 gives a more detailed overview of the interaction between our trajectory estimation and scoring. Throughout the sequence examined roughly 200 merge-split situations, of varying complexity, are resolved. Figure 7 shows a few results obtained. Solving these trajectories in this non-causal exhaustive manner proves to be very reliable.

Once all the possible simple merge and split situations have been solved, it is interesting to see how frequently a player is assigned to a *player track*. Figure 8 gives an overview of this information for our football game clip. The tracks are fairly evenly distributed throughout and are mainly only significantly interrupted by the major congestion scenarios of corner kicks and goals. One player "Left Midfield" is significantly less frequently in a player track. He is on the side of the field furtherest from the camera. Thus we see the effect of our lack of resolution on that side. It must be noted, though, that during the periods when a player is not assigned to a player track, he is assigned to a multiple player track. Given the interaction graph and player track identities, it is possible to ascertain the identities of the players in a multiple player track. The next sections are devoted to recreating the graph in figure 8(a) automatically.

5 Clustering Player Tracks

At this stage we have resolved as many of the simple split and merge situations as possible using continuity of appearance and motion of the player tracks. There are, however, other features that can be used to associate the identity of player tracks. For example in a football game, a player's identity can be frequently obtained by his relative position to his team-mates. The most obvious example of this is the goal-keeper, who is behind all his teammates. This section describes a feature vector built to encode this relative spatial information. A straight forward clustering regime, based on this feature vector, is then explained. The clustering aims to find clusters containing player tracks of the same identity. This allows for the identity of temporally separated player tracks to be linked and thus defines re-initialization points for each player throughout the game.

5.1 Player Track Feature Vector

The specific feature vector calculated to describe a player track is, of course, biased by the application domain. What follows is a football dependent feature vector. Though the same methodology could be applied to other applications.

Let ${}^A\mathcal{X}_t = \{{}^A\mathbf{x}_t^i\}_{i=1}^{11}$ be the x, y-coordinates of the team A players at frame t. For each $\mathbf{x}_t^i$ (dropping the A superscript for brevity) we construct a 4-dimensional vector $\mathbf{v}_t^i = (r_t^i, l_t^i, f_t^i, b_t^i)$ recording the number of players to the right, left, in front and behind player i, subject to a margin $\epsilon > 0$. As there are eleven players on each team $r_t^i + l_t^i \leq 10$ and $f_t^i + b_t^i \leq 10$. Thus there are 66^2 distinct possible v_t^i. To reduce this number the range from 0 to 10 is quantized into 6 bins $\{0,1\}, \{2,3\}, \cdots, \{9,10\}$. This quantization results in 21^2 distinct v_t^i. Each v_t^i is assigned an index, $\mathrm{id}(v_t^i)$ between 0 and 440.

From the subset of ellipses of $\mathcal{E}_{t_i+j}$ labelled as team A we can estimate ${}^A\mathcal{X}_{t_i+j}$. These computations take place once the players' feet positions from each ellipse have been estimated and transformed to a rectified version of the pitch. Consider the player track, $\mathcal{T}_i = \mathcal{T}(t_i, n_i, \mathbf{k}^i)$. The relative spatial arrangement of this player during the track is encoded by the v-vector for each of its ellipses:

$$\mathcal{V}_{\mathcal{T}_i} = (id(v_{t_i}^{k_0^i}), id(v_{t_i+1}^{k_1^i}), \cdots, id(v_{t_i+n_i-1}^{k_{n_i-1}^i})) \tag{17}$$

Histogram $\mathcal{V}_{\mathcal{T}_i}$ to define a feature vector $\mathbf{f}_{\mathcal{T}_i} \in [0,1]^{441}$ with jth entry

$$\mathbf{f}_{\mathcal{T}_i}(j) = \frac{1}{n_i}\sum_{s=1}^{n_i} \delta_{j, id(v_{t_s}^{k_s^i})}. \tag{18}$$

The distance between two tracks is then defined as $D(\mathcal{T}_i, \mathcal{T}_l) = \|\mathbf{f}_{\mathcal{T}_i} - \mathbf{f}_{\mathcal{T}_l}\|^2$.

5.2 Clustering Procedure

Let $\mathcal{U} = \{\mathcal{T}_i\}_{i=1}^K$ be the set of team A trajectories with length $>$ 10 seconds. This limit is chosen to ensure each trajectory has a reliably estimated feature vector. The goal is to partition $\mathcal{U}$ into L, 11 for football, clusters $\mathcal{C} = \{C_l\}$ s.t. each cluster corresponds exactly to one player. This partition is subject to the condition that one person can only be in one place at one time. Thus temporally overlapping trajectories cannot be assigned to the same cluster. An indicator function $\gamma(\mathcal{T}_i, \mathcal{T}_j)$ is set to 1 if $\mathcal{T}_i$ and $\mathcal{T}_j$ temporally overlap and 0 otherwise.

Cluster Initialization. We have no explicit model for each player. Therefore we rely upon un-supervised clustering. This necessitates proceeding carefully, especially initially. We want to ensure finding representative members for each cluster from which we can grow. For football, the longer a trajectory the more likely it is to incorporate the state of a player from several team formations. The clustering thus occurs in two stages - initialize the clusters and then expand them. The first stage initializes the clusters by examining only temporally very long trajectories which are fortuitously nicely separated in our chosen feature space. Explicit details of the algorithm are given in figure 9, essentially the algorithm finds compact clusters starting from tracks of decreasing temporal length. The results of applying this algorithm to the football data are displayed in figure 10. Thirteen clusters are found and each cluster contains only tracks of one identity.

Cluster Growing. The second stage of the clustering process involves expanding the initial clusters to include the other trajectories of non-trivial length and merging clusters when possible and necessary to reach the expected number of 11. To adequately describe the secondary clustering procedure some notation and concepts are now introduced. A temporally dependent subset, $a \subset \mathcal{U}$, is a subset in which every pair of member tracks temporally overlap. Then define $\mathcal{S}(\mathcal{U})$ as the set containing all such subsets of $\mathcal{U}$. An assignment, P_a, from a set

Data: A set of player trajectories $\{\mathcal{T}_{i_j}\}_{j=1}^{K_0} \subset \mathcal{U}$ with temporal lengths $n_{i_j} \geq 1000$.

Constraints: All members of a cluster are within a distance $\epsilon > 0$ of each other. No two members of a cluster can temporally overlap.

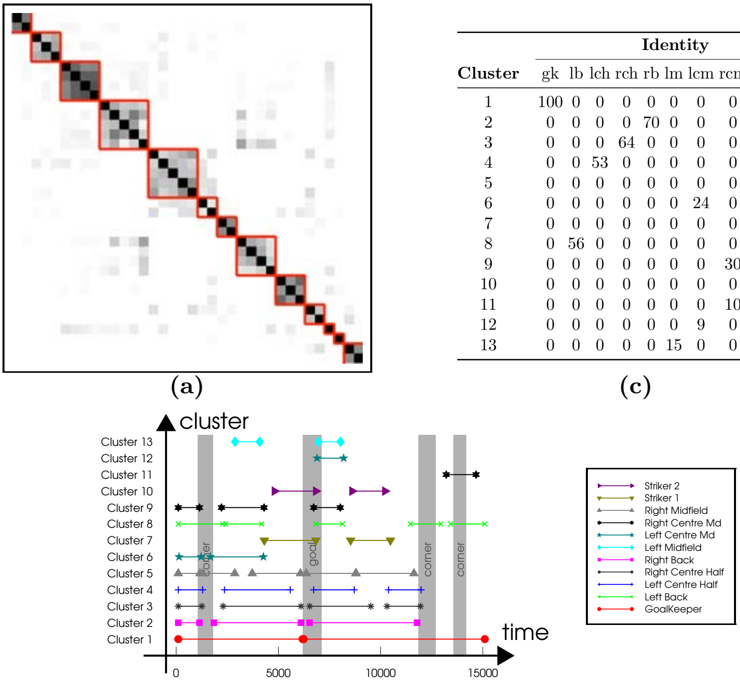
Algorithm: Let the set of clusters $\mathcal{C} = \emptyset$ and the set of unexplained trajectories $\mathcal{U}' = \{\mathcal{T}_{i_j}\}_{j=1}^{K_0}$. Initialize the counter variables $k = 0, l = 0$.

```
while $k \leq K_0$
  1. Choose $\mathcal{T}_{i_0}$ the longest track in $\mathcal{U}'$.
  2. Set $C_l = \{\mathcal{T}_{i_0}\}$.
     while $|C_l|$ is increasing
       – Find the longest track in $\mathcal{T}_{i_j} \in \mathcal{U}'$ such that

             $\gamma(\mathcal{T}_{i_j}, \mathcal{T}_k) = 0$ & $D(\mathcal{T}_{i_j}, \mathcal{T}_k) < \epsilon \quad \forall \mathcal{T}_k \in C_l$.

       – If such a $\mathcal{T}_{i_j}$ exists, set $C_l = C_l \cup \mathcal{T}_{i_j}$.
     end
  3. Set $\mathcal{C} = \mathcal{C} \cup C_l$, $\mathcal{U}' = \mathcal{U}' \backslash C_l$, $l = l + 1$ and $k = k + |C_l|$.
end
```

Fig. 9. The initial clustering algorithm



(a)

	Identity										
Cluster	gk	lb	lch	rch	rb	lm	lcm	rcm	rm	s1	s2
1	100	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	70	0	0	0	0	0	0
3	0	0	0	64	0	0	0	0	0	0	0
4	0	0	53	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	68	0	0
6	0	0	0	0	0	0	24	0	0	0	0
7	0	0	0	0	0	0	0	0	0	30	0
8	0	56	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	30	0	0	0
10	0	0	0	0	0	0	0	0	0	0	24
11	0	0	0	0	0	0	0	10	0	0	0
12	0	0	0	0	0	0	9	0	0	0	0
13	0	0	0	0	0	15	0	0	0	0	0

(c)

(b)

Fig. 10. **(a)** The distance between every pair of player tracks >40 sec is shown. Ordering is according to the clusters found by the initial clustering. Darker values indicate smaller distances. **(b)** This graph displays the temporal extent and true identity of the player tracks in each cluster. The legend shows the color and symbol representing each identity. **(c)** This confusion table summarizes the homogeneity of the identities for a cluster's temporal extent. Each entry is the sum of the temporal lengths of the player tracks in a cluster of one identity, shown as a percentage of the total sequence time.

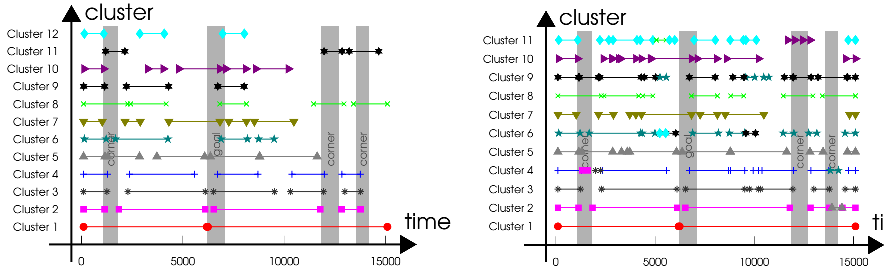
$a \in \mathcal{S}(\mathcal{U})$ is a one-to-one mapping $P_a : \{1, \cdots, |a|\} \rightarrow \{1, \cdots, L\}$ s.t. the ith track of a is assigned to cluster $P_a(i)$. P_a is valid if for each $\mathcal{T}_{a_i} \in a$

$$\exists \mathcal{T}_l \in C_{P_a(i)} \text{ s.t. } D(\mathcal{T}_l, \mathcal{T}_{a_i}) < \epsilon_1 \quad \& \quad \mathcal{T}_l \in C_{P_a(i)} \;\Rightarrow\; \gamma(\mathcal{T}_{a_i}, \mathcal{T}_l) = 0 \qquad (19)$$

where $\epsilon_1 > 0$. The cost of such an assignment is

$$\text{Sc}(P_a) = \sum_{i=1}^{|a|} \min_{\mathcal{T}_l \in C_{P_a(i)}} D(\mathcal{T}_l, \mathcal{T}_{a_i}). \qquad (20)$$

In essence the algorithm finds the valid assignments for temporally dependent subsets and chooses the assignment with least cost. Finding the best fit with respect to a temporal dependent subset offers greater robustness to using a greedy algorithm on individual tracks and is still computational feasible.



	Identity										
Cluster	gk	lb	lch	rch	rb	lm	lcm	rcm	rm	s1	s2
1	100	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	77	0	0	0	0	0	0
3	0	0	0	69	0	0	0	0	0	0	0
4	0	0	59	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	68	0	0
6	0	0	0	0	0	0	38	0	0	0	0
7	0	0	0	0	0	0	0	0	0	47	0
8	0	56	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	30	0	0	0
10	0	0	0	0	0	0	0	0	0	0	42
11	0	0	0	0	0	0	0	22	0	0	0
12	0	0	0	0	0	22	0	0	0	0	0

(a) Tracks > 750 frames

	Identity										
Cluster	gk	lb	lch	rch	rb	lm	lcm	rcm	rm	s1	s2
1	100	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	81	0	0	0	3	0	0
3	0	0	0	76	0	0	0	0	0	0	0
4	0	0	68	2	2	0	3	0	0	0	0
5	0	0	0	0	0	0	0	0	77	0	0
6	0	0	0	0	0	2	52	7	0	0	0
7	0	0	0	0	0	0	0	0	0	51	0
8	0	64	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	7	65	0	0	0
10	0	0	0	0	0	0	0	0	0	0	53
11	0	3	0	0	0	41	0	0	0	0	5

(b) Tracks > 250 frames

Fig. 11. Cluster growing results. The graphs and tables have the same format as figure 10. Column **(a)** shows the results when tracks of length > 750 are added to the initial clusters. In this case, homogeneity of identity in the clusters is maintained. However, when tracks of shorter length are added errors begin to occur, see column **(b)**. The columns of the right confusion table sum to the percentages displayed in figure 8 (b).

Results of applying the clustering algorithm are shown in figure 11. The left column shows the results of including the tracks with temporal lengths between 750 and 1000 frames and the right column the additional tracks over 250 frames. Results are good. Errors in the clustering begin to appear with the addition of the shorter length tracks. Most of the errors occur at the major events when the team switches between different formations. This would indicate that our feature vector should be extended to take the overall team formation into account.

6 Conclusions and Discussion

This paper presents an approach to multi-target tracking and labelling that is viable on long sequences with many targets, assuming there are no real-time constraints. At each stage the reliable information is extracted and built upon. Initially we find the trajectories when players are isolated, extend and concatenate these trajectories when possible by resolving merge-split situations. From these trajectories a large graph summarizing the player interactions throughout the game is built. The identities of the found, but temporally spread, trajectories are linked using a two-stage clustering scheme. This sets re-initialization points for a player's identity throughout the sequence. In combination with our interaction graph this gives us, potentially, an estimation of each player's position throughout the sequence, with a varying degree of accuracy depending on whether at a time instant he is assigned to a *player track* or *multiple player track*.

The methods are scalable to a whole game. We anticipate similar results, if not better, could be obtained with more data. The feature vector may require updating to cope with the different possible team formations. The labelling of the shorter tracks may also require greater sophistication, taking into account the graph structure and ensuring there is a path between every member of a cluster. One word of caution though is that a fairly robust background subtraction process underpins this work. This is made possible by our wide screen video and sports environment. A more probabilistic approach to the extraction of the initial trajectories may allow more cluttered environments to be considered.

Acknowledgments. This work was supported by the Swedish Foundation for Strategic Research funded project VISCOS. The authors would like to thank Eric Hayman for providing the wide-screen video.

References

1. Khan, Z., Balch, T., Dellaert, F.: An mcmc-based particle filter for tracking multiple interacting targets. In: European Conference on Computer Vision. (2004)
2. Gelgon, M., Bouthemy, P., Le Cadre, J.: Recovery of the trajectories of multiple moving objects in an image sequence with a pmht approach. J. Image & Vision Computing **23** (2005) 19–31
3. Xu, M., Orwell, J., Jones, G.: Tracking football players with multiple cameras. In: IEEE International Conference on Image Processing. (2004)
4. Iwase, S., Saito, H.: Parallel tracking of all soccer players by integrating detected positions in multiple view images. In: ICPR. (2004) 751–754
5. Vermaak, J., Doucet, A., Perez, P.: Maintaining multi-modality through mixture tracking. In: International Conference on Computer Vision. (2003)
6. Okuma, K., Taleghani, A., De Freitas, N., Little, J.J., Lowe, D.G.: A boosted particle filter: Multitarget detection and tracking. In: ECCV. (2004)
7. Needham, C., Boyle, R.: Tracking multiple sports players through occlusion, congestion and scale. In: BMVC. (2001)
8. Figueroa, P., Leite, N., Barros, R., Cohen, I., Medioni, G.: Tracking soccer players using the graph representation. In: ICPR. (2004) 787–790
9. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: Conference on Computer Vision and Pattern Recognition. (1999)