Chapter 12
Representing Team Behaviours from Noisy Data Using Player Role

Alina Bialkowski, Patrick Lucey, Peter Carr, Sridha Sridharan and Iain Matthews

Abstract Due to their unobtrusive nature, vision-based approaches to tracking sports players have been preferred over wearable sensors as they do not require the players to be instrumented for each match. Unfortunately however, due to the heavy occlusion between players, variation in resolution and pose, in addition to fluctuating illumination conditions, tracking players continuously is still an unsolved vision problem. For tasks like clustering and retrieval, having noisy data (i.e. missing and false player detections) is problematic as it generates discontinuities in the input data stream. One method of circumventing this issue is to use an occupancy map, where the field is discretised into a series of zones and a count of player detections in each zone is obtained. A series of frames can then be concatenated to represent a set-play or example of team behaviour. A problem with this approach though is that the compressibility is low (i.e. the variability in the feature space is incredibly high). In this paper, we propose the use of a bilinear spatiotemporal basis model using a *role representation* to clean-up the noisy detections which operates in a low-dimensional space. To evaluate our approach, we used a fully instrumented field-hockey pitch with 8 fixed high-definition (HD) cameras and evaluated our approach on approximately 200,000 frames of data from a state-of-the-art real-time player detector and compare it to manually labeled data.

A. Bialkowski (✉) · S. Sridharan
Queensland University of Technology, Brisbane, QLD, Australia
e-mail: a.bialkowski@connect.qut.edu.au

S. Sridharan
e-mail: s.sridharan@qut.edu.au

A. Bialkowski · P. Lucey · P. Carr · I. Matthews
Disney Research, Pittsburgh, PA, USA

P. Lucey
e-mail: patrick.lucey@disneyresearch.com

P. Carr
e-mail: peter.carr@disneyresearch.com

I. Matthews
e-mail: iainm@disneyresearch.com

T.B. Moeslund et al. (eds.), *Computer Vision in Sports*,
Advances in Computer Vision and Pattern Recognition,
DOI 10.1007/978-3-319-09396-3_12

12.1 Introduction

As the sophistication of analysis increases in professional sport, more organisations are looking at using player tracking data to obtain an advantage over their competitors. For sports like field-hockey, the dynamic and continuous nature makes analysis extremely challenging as game-events are not segmented into discrete plays, the speed of play is very quick (e.g. the ball can move at 125 km/h), and the size of the field is very large, with each player free to occupy any area at any time. A common approach to this problem is to use each player's x, y position in every frame to recognise team events [19, 20, 23]. However, as reliably tracking players in this environment over relatively long periods of time (i.e. >1 min) remains an unsolved computer vision problem, often large amounts of tracking data contains "holes" or "gaps" making analysis very difficult.

For tasks such as automatic event annotation (e.g. goals, corners, free-kicks), representations that describe the global pattern of team behaviours such as team-centroids or occupancy can be utilised on noisy detections. While these *macroscopic* representations can pick up on the global patterns, specific information such as individual player behaviours may be ignored which could be important for more specific events or retrieval tasks. As such, a *microscopic* representation (i.e. continuous tracks of each player) is preferred but this requires human intervention for long tracks. An example of this is shown in Fig. 12.1.

As player motion and position (i.e. proximity to teammates and opponents) is heavily linked to game-context and where the action on the field is taking place, these contextual features can be used to fill in the gaps of missed tracks caused by missed or false detections. In team sports, an important contextual feature is characterised by a *formation*: a coarse spatial structure which the players maintain over the course of the match. Additionally, player movements are governed by physical limits, such as acceleration, which makes trajectories smooth over time. These two observations suggest significant correlation (and therefore redundancy) in the spatiotemporal signal of player movement data. A core contribution of this work is to recover a low-dimensional approximation for a time series of player locations. The compact representation is critical for understanding team behaviour. First, it enables the recovery of a true underlying signal from a set of noisy detections. Second, it allows for efficient clustering and retrieval of game events.

A key insight of this work is that even perfect tracking data is not sufficient for understanding team behaviour, and an appropriate representation is necessary. A formation implicitly defines a set of *roles* or individual responsibilities which are then distributed amongst the players by the captain or coach. In dynamic games like field hockey, it may be opportunistic for players to swap roles (either temporarily or permanently). As a result, when analysing the strategy of a particular game situation, players are typically identified by the role they are currently playing and not necessarily by an individualistic attribute like name. In this paper, we present a *role representation* which provides a more compact representation compared to player identity, and allows us to use subspace methods such as the bilinear spatiotemporal basis model [3] to "denoise" noisy detections (which is common from a vision system).

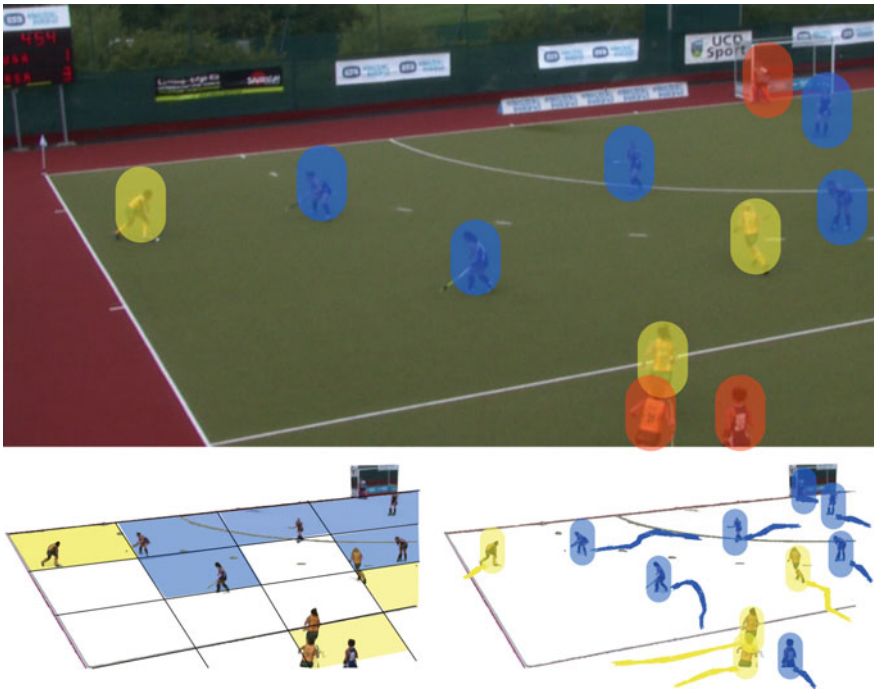

Fig. 12.1 (*Top*) Detecting and tracking players over long-periods of time is challenging and often results in missed detections (*highlighted in red*) and false detections. (*Bottom*) In this paper, we evaluate two representations which are robust to false and missed detections: (1) an occupancy map (*left*), which is formed by quantising the field into discrete areas, and (2) a bilinear spatiotemporal model, which can be used to estimate continuous player tracks (*right*) from detection data and compactly represent spatiotemporal patterns

To enable this research we used player detection data captured via 8 fixed high-definition (HD) cameras, across seven complete field-hockey matches (over 8 h of match data for each camera). We utilise a state-of-the-art real-time player detector [9] to give player positions at every frame, affiliate detection results into teams using a colour histogram model, and then compare two approaches for representing the team behaviours: an occupancy map representation and a bilinear spatiotemporal basis model, which models the movement of players by role.

12.2 Related Work

Due to the host of military, surveillance and sport applications, research into recognising group behaviour has recently increased dramatically. Outside of the sports realm, most of this work has focussed on dynamic groups, where individual agents can leave and join groups over the period of observation. An initial approach

was to recognise the activities of individual agents and then combine these to infer group activities [5]. Sukthankar and Sycara [33, 34] recognised group activities as a whole but pruned the size of possible activities by using temporal ordering constraints and agent resource dependencies. Sadilek and Kautz [30] used GPS locations of multiple agents in a "capture the flag" game to recognise low-level activities such as approaching and being at the same location. All of these works assume that the position and movements of all agents are known, and that all behaviours can be mapped to an activity within the library. Recently, Zhang et al. [37] used a "bag of words" and Support Vector Machine (SVM) approach to recognise group activities on the Mock Prison dataset [10].

Sports related research mostly centres on low-level activity detection with the majority conducted on American Football. In the seminal work by Intille and Bobick [18], they recognised a single football play *pCurl51*, using a Bayesian network to model the interactions between the players trajectories. Li et al. [24], modelled and classified five offensive football plays (dropback, combo dropback, middle run, left run, right run). Siddiquie et al. [31], performed automated experiments to classify seven offensive football plays using a shape (HoG) and motion (HoF) based spatio-temporal features. Instead of recognising football plays, Li and Chellapa [23] used a spatio-temporal driving force model to segment the two groups/teams using their trajectories. Researchers at Oregon State University have also done substantial research in the football space [15, 16, 32] with the goal of automatically detecting offensive plays from a raw video source and transferring this knowledge to a simulator. For soccer, Kim et al. [20] used the global motion of all players in a soccer match to predict where the play will evolve in the short-term. Beetz et al. [6] developed the *automated sport game models* (ASPOGAMO) system which can automatically track player and ball positions via a vision system. Using soccer as an example, the system was used to create a heat-map of player positions (i.e. which area of the field did a player mostly spend time in) and also has the capability of clustering passes into low-level classes (i.e. long, short etc.), although no thorough analysis was conducted due to a lack of data. In basketball, Perse et al. [28] used trajectories of player movement to recognise three type of team offensive patterns. Morariu and Davis [27] integrated interval-based temporal reasoning with probabilistic logical inference to recognise events in one-on-one basketball. Hervieu and Bouthemy [14] also used player trajectories to recognise low-level team activities using a hierarchical parallel semi-Markov model.

It is worth noting that an enormous amount of research interest has used broadcast sports footage for video summarisation in addition to action, activity and highlight detection [6, 12, 13, 17, 22, 25, 26, 36], but given that these approaches are not automatic (i.e. the broadcast footage is generated by humans) and that the telecasted view captures only a portion of the field, analysing groups has been impossible because some individuals are normally out of frame. Although similar in spirit to the research mentioned above, our work differs as: (1) we rely only on player detections rather than tracking, and (2) we compare across many matches.

12.3 Detection Data

12.3.1 Field-Hockey Test-Bed

In this work, we investigate the behaviours of several international field-hockey teams from player tracking data. To enable this research, we recorded video footage from a recent field-hockey tournament using eight stationary HD cameras which provide complete coverage of the 91.4 × 55.0 m playing surface, as displayed in Fig. 12.2.

12.3.2 Player Detection and Team Affiliation

For each camera, player image patches are extracted using a real-time person detector [9], which detects players by interpreting background subtraction results in terms of 3D geometry, where players are coarsely modelled as cylinders of height 1.8 m. This equates to 40–100 pixels height in the image depending on the distance

Fig. 12.2 View of the field-hockey pitch from the 8 fixed HD cameras

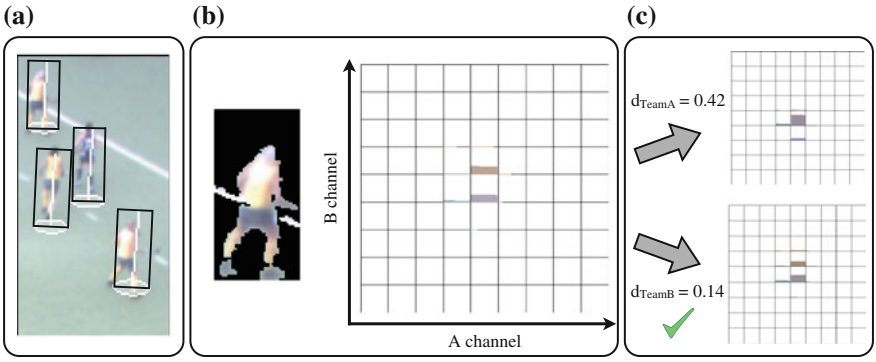

Fig. 12.3 **a** We detect players using a real-time person detector, and **b** represent the colours of each image patch using histograms of the foreground pixels in LAB colour space. **c** An image patch is then assigned to the closer of the two team histogram models (or "other" if the distance to each model exceeds a threshold)

from the camera, so for scale invariance, we normalise the image patches to a fixed size of 90×45 pixels. The image patches are then classified into teams using colour histograms of the foreground pixels, as illustrated in Fig. 12.3. The LAB colour space is used for representing the colours of each image patch, ignoring the luminance channel as it is affected by illumination changes. Nine bins are used for each dimension, and the histograms are normalised to sum to one.

Models for the two teams are learnt using k-means clustering from a training set of approximately 4,000 training histograms, and the Bhattacharyya coefficient is used as the comparison metric. A detected image patch is then classified to the closer of the two team models, or if it falls outside a threshold, it is classified as "others" (i.e. noise, referees, goalies). In our dataset, teams wear contrasting colours, so colour histograms are sufficient for distinguishing between the two teams. Detections from the eight cameras are then aggregated by projecting the player positions to field coordinates using each camera's homography, and merging player detections based on proximity (Fig. 12.4).

The performance of the detector and team classification compared to ground truth annotated frames using precision and recall metrics is shown in Table 12.1. From this table, it can be seen that while recall is high, the team classification has quite low precision in some matches. The poor performance is mainly attributed to non-team-players (referees, goalies, and false-positive player detections caused by background clutter) being misclassified into one of the teams, as they contain a combination of team colours. A more sophisticated representation could be used for modelling the teams as well as non-team image patches, and online learning of the colour models to adapt with changes in illumination would further improve results. From these results, it is evident that our team behaviour representation must be able to deal with a high degree of noise.

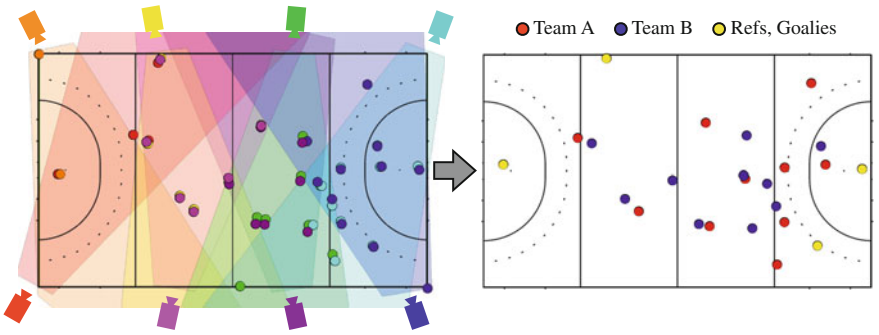

Fig. 12.4 (*Left*) We detect players in each camera using a real-time person detector and project these into real world co-ordinates. (*Right*) We then classify the detections into one of the two teams (or others) and aggregate the detections from each camera to extract the state of the game at each time instant

Table 12.1 Precision and recall values of the player detector and team affiliation classifier, after aggregating all cameras

Match code	No. of Frames	Precision			Recall		
		Detector (%)	Team A (%)	Team B (%)	Detector (%)	Team A (%)	Team B (%)
10-USA-RSA-1	14,352	81.1	67.2	77.7	89.0	98.3	98.4
24-JPN-USA-1	20,904	89.5	91.7	90.0	87.5	95.2	97.4
24-JPN-USA-2	7,447	85.8	72.4	79.7	90.0	97.6	97.0

12.4 Modelling Team Behaviours from Detections

An intuitive representation of team behaviours in sports would be to track all players (maintaining their identity) and the ball. For field-hockey, this would result in a 42 dimensional signal per frame (i.e. 21 objects with x and y coordinates—10 field players excluding the goalie $\times$ 2 teams, and the ball). However, since we cannot reliably and accurately track the player and ball over long durations, an alternative is to represent the match via player detections.

Player detection data can be modelled as a series of observations $\mathcal{O}$, where each observation consists of an (x, y) ground location, a timestamp t, and a team affiliation estimate $\tau \in \{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$. At any given time instant t, the set of detected player locations $\mathcal{O}_t = \{x_A, y_A, x_B, y_B, \ldots\}$ is of arbitrary length because some players may not have been detected and/or background clutter may have been incorrectly classified as a player. Therefore, the number of player detections at any given frame is generally not equal to the actual number of players $2P$, where $P = 10$ players per team.

By representing the detections at each frame, we overcome the issue of tracking, but as a consequence we remove the player identity component of the signal and need another method to maintain feature correspondences. We propose to employ

an *occupancy map* descriptor, which is formed by breaking the field into a series of spatial bins and counting the number of players that occupy each of the bins.

12.4.1 Team Occupancy Maps

The team occupancy descriptor, $\mathbf{x}_t^o$, is a quantised occupancy map of the player positions on the field for each team represented at time t. Given we have the locations of players from the player detector system and have assigned team affiliation, an occupancy map can be constructed for each frame by quantising the $91.4 \times 55.0\,\text{m}$ field into K bins, and counting how many player detections for that team fall within each location. The dimensionality of the formation descriptor is equal to twice the number of bins (i.e. $K \times 2$) so that both teams A and B are accounted for, resulting in $\mathbf{x}_i^o = [a_1, \ldots, a_K; b_1, \ldots, b_K]$, where a_k and b_k are the player counts in bin k for teams A and B respectively. Such an occupancy map can then be used to represent team activities by concatenating frames.

Depending on the level of complexity of the activity that we wish to recognise, we can use varying descriptor sizes (coarse/fine). We evaluate five different descriptor sizes: $K = 2(2 \times 1)$, $K = 8(4 \times 2)$, $K = 32(8 \times 4)$, $K = 135(15 \times 9)$ and $K = 540(30 \times 18)$, with examples illustrated in Fig. 12.5. The different quantisations represent how much tolerance there is in player's positions (e.g. in 15×9 quantisation, each player is assigned to an area of approximately $6\,\text{m}^2$).

12.4.2 Recognising Team Activities

To evaluate the different occupancy map representations, we conducted a series of isolated activity recognition experiments. We use the occupancy maps to recognise five activities, corresponding to important game states in field-hockey, shown in Fig. 12.6. As these activities coincide with a single event (e.g. the ball crossing the out line, or a goal being scored), they do not have distinct onset and offset times.

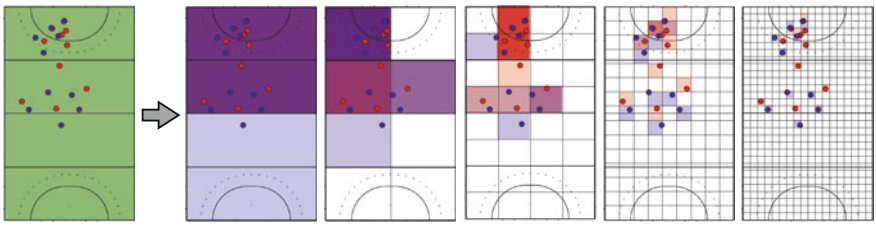

Fig. 12.5 Example team occupancy maps for different descriptor sizes

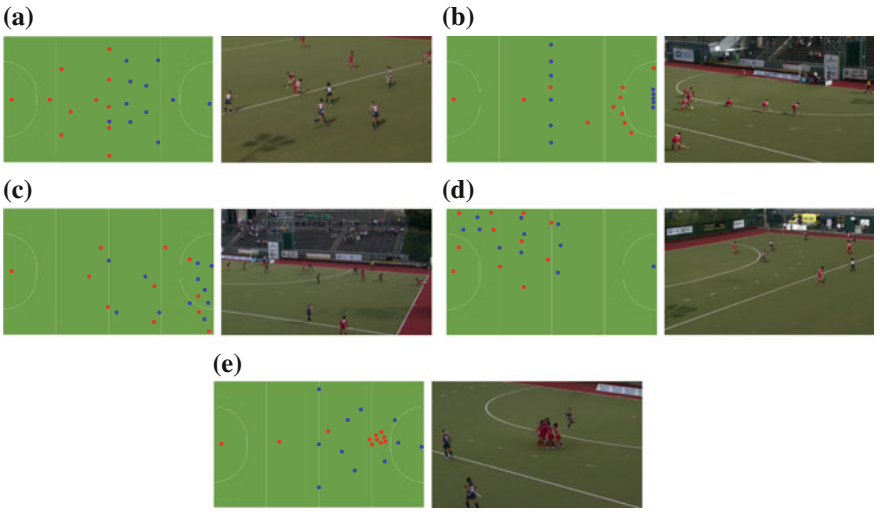

Fig. 12.6 Diagrams and examples of structured plays that occur in field-hockey. **a** Faceoff. **b** Penalty corner. **c** Long corner. **d** Defensive corner. **e** Goal

To account for this, we used the event as the start of the activity and went forward 10 s as the offset time, which gave us a series of 10 s play clips.

Since an activity can occur for either team, we compare the template descriptors in both orientations ($\mathbf{x}^o = [\mathbf{a}, \mathbf{b}]^T$, and $\mathbf{x}^o = [\mathbf{b}_{rot}, \mathbf{a}_{rot}]^T$, where $\mathbf{a}_{rot}$ represents a rotation of the field by 180° for team a's formation descriptor, so that the new descriptor is given by $\mathbf{a}_{rot}[k] = \mathbf{a}[K + 1 - k]$, for $k = 1, 2, \ldots, K$). We calculate the distance to the template in both orientations, and take the minimum as the distance measure.

Seven full matches (corresponding to over 8 hours of game play), were annotated with the 5 activities of interest: face-offs, penalty corners, goals, long corners and defensive corners as shown in Table 12.2. The annotated activities were split into testing and training sets using a leave-one-out cross-validation strategy, where one match half was used for testing and the remaining halves for training. We used a k-Nearest Neighbour classification approach, taking the mode activity label of the closest k examples in the training set, using L_2 as our distance measure. Confusion matrices using $k = 10$ are presented in Fig. 12.7.

Most activities are well recognised, however goals are often misclassified as the other activities because they are less structured, with a lot of variability possible. Defensive corners and long corners are sometimes confused with one another as the main difference is the team which maintains possession, which is not discernible from the occupancy descriptors. The best accuracy was achieved using an 8×4 descriptor with an accuracy of 78.2 %. Quantising at a finer level beyond this, resulted in a slightly reduced accuracy, which can be explained by players not aligning to the exact locations in the training activity templates, due to variability in activities (and our distance metric only compares corresponding field locations between occupancy

Table 12.2 Frequency of the annotated activities in each match half

	Face off	Penalty corner	Goal	Long corner		Defensive corner	
				(L)	(R)	(L)	(R)
1-JPN-USA-1	3	2	2	11	5	4	4
1-JPN-USA-2	2	6	1	4	10	7	3
2-RSA-SCO-1	2	4	2	11	4	3	3
2-RSA-SCO-2	3	9	2	3	12	4	3
5-USA-SCO-1	3	4	2	7	4	1	7
5-USA-SCO-2	3	8	2	3	3	2	2
9-JPN-SCO-1	2	4	2	8	7	5	2
9-JPN-SCO-2	1	1	0	10	10	6	0
10-USA-RSA-1	5	9	5	5	5	8	0
10-USA-RSA-2	6	4	5	6	7	4	1
23-ESP-SCO-1	3	4	2	7	6	1	1
23-ESP-SCO-2	3	7	2	9	5	2	1
24-JPN-USA-1	4	3	3	9	6	5	1
24-JPN-USA-2	2	2	1	5	9	7	6
Total	42	67	31	98	93	59	34

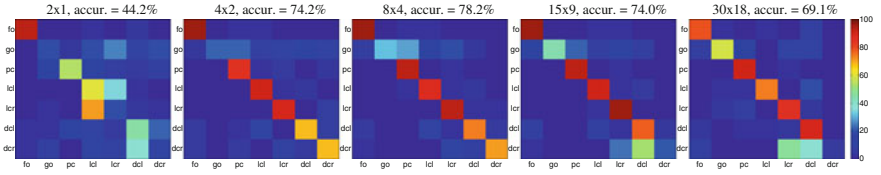

Fig. 12.7 Confusion matrices for isolated activity recognition using different occupancy map descriptor sizes

maps). A more coarse descriptor is able to represent the activity with tolerance for player position variations. This indicates that the annotated activities can be described by the distribution of the players on a relatively macroscopic scale rather than the exact positions approximated with the finer descriptors.

Despite their simplicity, it is evident that occupancy maps can be used to recognise important game states in the presence of noise and without any tracking information. However, if we wish to recognise the fine behaviours of teams (i.e. at the level of individual player behaviours), an occupancy map representation requires a very high dimensionality feature vector (e.g. a grid of 30×18 requires 540 feature dimensions per frame to represent player locations to a precision of ~ 3 m^2). In addition, when modelling longer term behaviours, occupancy map descriptors are not very compressible in the temporal domain, because they do not directly model player movements (which are smooth) but occupancies in different zones of the field, which are discrete and do not vary smoothly or predictably in time, particularly with noisy detections.

In order to model individual behaviours in team sports compactly, we need a method to clean-up noisy detections and a representation which exploits the high degree of correlation between players. Player tracks could allow this, but we must overcome the issue of noisy detections (i.e. missed and false detections).

12.5 Representing Adversarial Movements

The task of tracking players across time is equivalent to generating a vector of ordered player locations $\mathbf{p}_t^\tau = [x_1, y_1, x_2, y_2, \ldots, x_P, y_P]^\mathsf{T}$ for each team τ from the noisy detections $\mathcal{O}_t$ at each time instant. The particular ordering of players is arbitrary, but must be consistent across time. Therefore, we refer to $\mathbf{p}_t^\tau$ as a *static labeling* of player locations. It is important to note that $\mathbf{p}_t^\tau$ is not simply a subset of $\mathcal{O}_t$. If a player was not detected, an algorithm must somehow infer the location of the unseen player.

We focus on generic team behaviours and assume any observed arrangement of players from team α could also have been observed for players from team β. As a result, there is a 180° symmetry in our data. For any given vector of player locations $\mathbf{p}_t^\tau$, there is an equivalent complement $\overleftrightarrow{\mathbf{p}}_t^\tau$ from rotating all (x, y) locations about the centre of the field and swapping the associated team affiliations.

12.5.1 Formations and Roles

In the majority of team sports, the coach or captain designates an overall structure or system of play for a team. In field hockey, the structure is described as a *formation* involving *roles* or individual responsibilities (see Fig. 12.8). For instance, the 5:3:2 formation defines a set of roles $\mathcal{R} = \{$left back (LB), right back (RB), left halfback

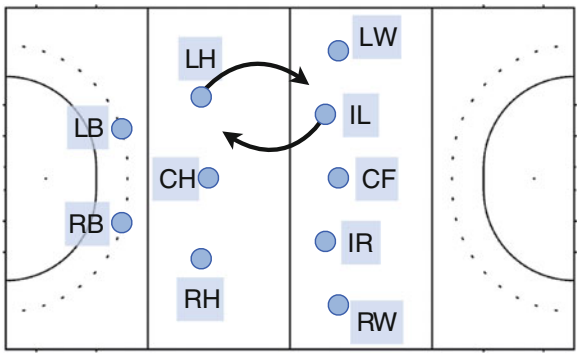

Fig. 12.8 The dynamic nature of the game requires players to switch roles and responsibilities on occasion, for example, the *left* halfback LH overlaps with the inside *left* IL to exploit a possible opportunity

Table 12.3 We manually labelled player location, identity and role at each frame for parts of four games from an international field-hockey tournament

Match code	No. of frames
10-USA-RSA-1	3,894
10-USA-RSA-2	8,839
24-JPN-USA-1	4,855
24-JPN-USA-2	7,418

(LH), center halfback (CH), right halfback (RH), inside left (IL), inside right (IR), left wing (LW), center forward (CF), right wing (RW)}. Each player is assigned exactly one role, and every role is assigned to only one player at any instant in time. Due to the dynamic nature of team sports, roles are generally not fixed and players will swap roles throughout a match, temporarily adopting the responsibilities of another player. Mathematically, assigning roles is equivalent to permuting the player ordering $\mathbf{p}_t^\tau$. We define a $P \times P$ permutation matrix $\mathbb{x}_t^\tau$ at time t which describes the players in terms of roles $\mathbf{r}_t^\tau$

$$\mathbf{r}_t^\tau = \mathbb{x}_t^\tau \mathbf{p}_t^\tau \tag{12.1}$$

By definition, each element $\mathbb{x}_t^\tau(i, j)$ is a binary variable, and every column and row in $\mathbb{x}_t^\tau$ must sum to one. If $\mathbb{x}_t^\tau(i, j) = 1$ then player i is assigned role j. In contrast to $\mathbf{p}_t^\tau$, we refer to $\mathbf{r}_t^\tau$ as a *dynamic labeling* of player locations.

Because the spatial relationships of a formation are defined in terms of roles (and not individualistic attributes like name) and players swap roles during the game, we expect the spatiotemporal patterns in $\{\mathbf{r}_1^\tau, \mathbf{r}_2^\tau, \ldots, \mathbf{r}_T^\tau\}$ to be more compact compared to $\{\mathbf{p}_1^\tau, \mathbf{p}_2^\tau, \ldots, \mathbf{p}_T^\tau\}$. Additionally, we expect a team to maintain its formation while moving up and down the field. As a result, position data $\tilde{\mathbf{r}}_t^\tau$ expressed relative to the mean (x, y) location of the team should be even more compressible. To test these conjectures, we manually tracked all players over 25,000 time-steps (which equates to $8 \times 25{,}000 = 200{,}000$ frames across 8 cameras), and asked a field hockey expert to assign roles to the player locations in each frame. A breakdown of the manually labelled data is given in Table 12.3.

For brevity, we explain the analysis in terms of roles $\mathbf{r}_t^\tau$ since the original player ordering $\mathbf{p}_t^\tau$ is just a special non-permuted case $\mathbb{x}_t^\tau = \mathbb{I}$. We ran PCA on the temporal data series produced by both teams $\{\mathbf{r}_1^\tau, \mathbf{r}_2^\tau, \ldots, \mathbf{r}_{25,000}^\tau, \overleftrightarrows{\mathbf{r}}_1^\tau, \overleftrightarrows{\mathbf{r}}_2^\tau, \ldots, \overleftrightarrows{\mathbf{r}}_{25,000}^\tau\}$. This was to measure how well the low-dimensional representation $\hat{\mathbf{r}}_t^\tau$ matches the original data $\mathbf{r}_t^\tau$ using the L_∞ norm of the residual $\Delta\mathbf{r} = \hat{\mathbf{r}}_t^\tau - \mathbf{r}_t^\tau$

$$\|\Delta\mathbf{r}\|_\infty = \max(\|\Delta\mathbf{r}(1)\|_2, \ldots, \|\Delta\mathbf{r}(P)\|_2) \tag{12.2}$$

where $\|\Delta\mathbf{r}(p)\|_2$ is the L_2 norm of the pth x and y components of $\Delta\mathbf{r}$. We chose the L_∞ norm instead of the L_2 norm because large deviations may signify very different formations, e.g. a single player could be breaking away to score. Figure 12.9 illustrates how both $\mathbf{p}_t^\tau$ and $\mathbf{r}_t^\tau$ are quite compressible on the training data. However, when we test on unseen data (with role labels), the dynamic role-based ordering $\mathbf{r}_t^\tau$

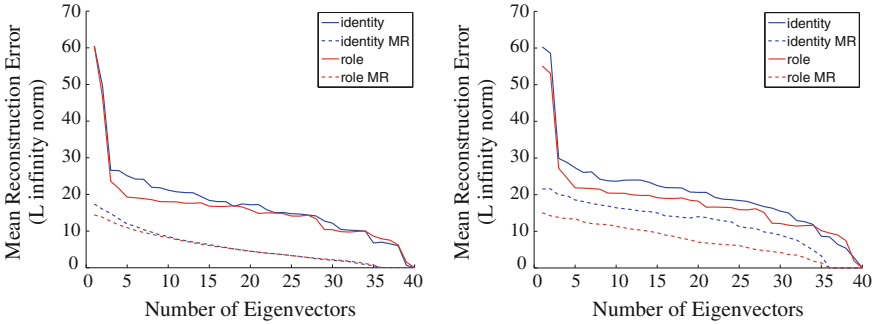

Fig. 12.9 Plot showing the reconstruction error as a function of the number of eigenvectors used to reconstruct the signal using the L_∞ norm for original and mean-removed features for both identity and role representations on training data (*left*) and unseen test data (*right*)

is much more compressible than the static ordering $\mathbf{p}_t^\tau$. Relative positions are more compressible than absolute positions in both orderings.

12.5.2 Incorporating Adversarial Behaviour

A player's movements are correlated not only to teammates but to opposition players as well. Therefore, we anticipate that player location data can be further compressed if the locations of players on teams A and B are concatenated into a single vector $\mathbf{r}_t^{AB} = [\mathbf{r}_t^A, \mathbf{r}_t^B]^\mathsf{T}$.

In Fig. 12.10, we show the mean formations for the identity and role representation. We can see that the role representation has a more uniform spread between the players, while the identity representation has a more crowded shape, which highlights the constant swapping of roles during a match. In terms of compressibility, Table 12.4 shows that using an adversarial representation gains better compressibility for both cases, and that using both a role and adversarial representation yields the most compressibility.

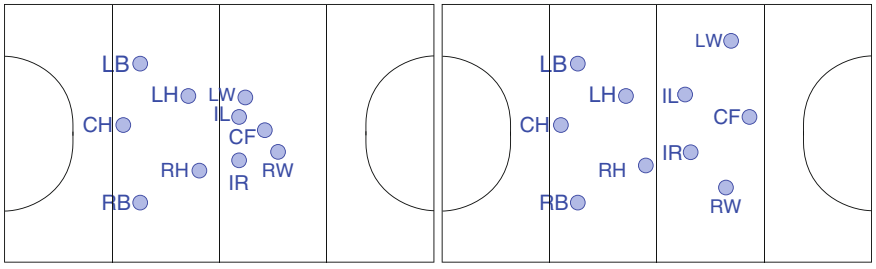

Fig. 12.10 Examples showing the difference between the mean formations using the: (*left*) identity and (*right*) role representations on one of the matches

Table 12.4 Showing the compressibility of different representations. Compressibility in this context refers to the percentage of features required to represent 95 % of the original signal

Representation	Compressibility	
	Identity (%)	Role (%)
Single team	30	25
Adversarial teams	20	**15**

12.6 Cleaning-Up Noisy Data

12.6.1 Spatiotemporal Bilinear Basis Model

The representation of time-varying spatial data is a well-studied problem in computer vision (see [8] for overview). Recently, Akhter et al. [3], presented a bilinear spatiotemporal basis model which captures and exploits the dependencies across both the spatial and temporal dimensions in an efficient and elegant manner, which can be applied to our problem domain. Given we have P players per team, we can form our role-based adversarial representation, $\mathbf{r}$, as a spatiotemporal structure $\mathbf{S}$, given $2P$ total players sampled at F time instances as

$$\mathbf{S}_{F\times 2P} = \begin{bmatrix} r_1^1 & \dots & r_{2P}^1 \\ \vdots & & \vdots \\ r_1^F & \dots & r_{2P}^F \end{bmatrix} \tag{12.3}$$

where r_j^i denotes the jth index within the role representation at the ith time instant. Thus, the time-varying structure matrix $\mathbf{S}$ contains $2FP$ parameters. This representation of the structure is an over parameterization because it does not take into account the high degree of regularity generally exhibited by motion data. One way to exploit the regularity in spatiotemporal data is to represent the 2D formation or shape at each time instance as a linear combination of a small number of shape basis vectors $\mathbf{b}_j$ weighted by coefficients ω_j^i as $\mathbf{s}^i = \sum_j \omega_j^i \mathbf{b}_j^T$ [7, 11]. An alternative representation of the time-varying structure is to model it in the trajectory subspace, as a linear combination of trajectory basis vectors, $\boldsymbol{\theta}_i$ as $\mathbf{s}_j = \sum_i a_i^j \boldsymbol{\theta}_i$, where a_i^j is the coefficient weighting each trajectory basis vector [1, 35]. As a result, the structure matrix can be represented as either

$$\mathbf{S} = \boldsymbol{\Omega}\mathbf{B}^T \quad \text{or} \quad \mathbf{S} = \boldsymbol{\Theta}\mathbf{A}^T \tag{12.4}$$

where $\mathbf{B}$ is a $P \times K_s$ matrix containing K_s shape basis vectors, each representing a 2D structure of length $2P$, and $\boldsymbol{\Omega}$, is an $F \times K_s$ matrix containing the corresponding shape coefficients ω_j^i; and $\boldsymbol{\Theta}$ is an $F \times K_t$ matrix containing K_t trajectory basis as its columns, and $\mathbf{A}$ is a $2P \times K_t$ matrix of trajectory coefficients. The number of shape basis vectors used to represent a particular instance of motion data is

$K_s \leq \min\{F, 2P\}$, and $K_t \leq \{F, 2P\}$ is the number of trajectory basis vectors spanning the trajectory subspace.

Both representations of $\mathbf{S}$ are over parameterisations because they do not capitalise on either the spatial or temporal regularity. As $\mathbf{S}$ can be expressed exactly as $\mathbf{S} = \boldsymbol{\Omega}\mathbf{B}^T$ and also $\mathbf{S} = \boldsymbol{\Theta}\mathbf{A}^T$, then there exists a factorization

$$\mathbf{S} = \boldsymbol{\Theta}\mathbf{C}\mathbf{B}^T \tag{12.5}$$

where $\mathbf{C} = \boldsymbol{\Theta}^T\boldsymbol{\Omega} = \mathbf{A}^T\mathbf{B}$ is a $K_t \times K_s$ matrix of spatiotemporal coefficients. This equation describes the bilinear spatiotemporal basis, which contains both shape and trajectory bases linked together by a common set of coefficients.

Due to the high degree of temporal smoothness in the motion of humans, a predefined analytical trajectory basis can be used without significant loss in representation. A particularly suitable choice of a conditioning trajectory basis is the Discrete Cosine Transform (DCT) basis, which has been found to be close to the optimal Principal Component Analysis (PCA) basis if the data is generated from a stationary first-order Markov process [29]. Given the high temporal regularity present in almost all human motion, it has been found that the DCT is an excellent basis for trajectories of faces [2, 3] and bodies [4]. Figure 12.11 shows that due to the highly structured nature of the game, and the fact that human motion over short periods of time is very simple, we can gain enormous dimensionality reduction especially in the temporal domain. From this, we can effectively represent 5 s plays with no more than $K_t = 3$ and $K_s = 33$ with a maximum error of less than 2 m. In terms of dimensionality reduction, this means we can represent temporal signals using $3 \times 33 = 99$ coefficients. For 5 s plays, this means a reduction of over 60 times. We found greater compressibility could be achieved on longer plays.

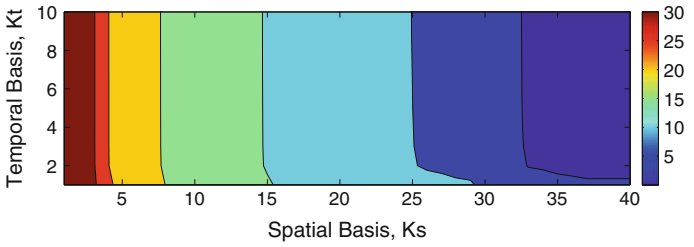

Fig. 12.11 Plot showing the mean reconstruction error of the test data as the number of temporal basis (K_t) and spatial basis (K_s) vary for 5 s plays (i.e. $K_{t\max} = 150$). We magnified the plot to show the first 10 temporal basis to highlight that only $K_t = 3$ is required to represent coarse player motion

12.6.2 The Assignment Problem

In the previous section, roles were specified by a human expert. We now address the problem of automatically assigning roles to an arbitrary ordering of player locations $\mathbf{p}_t^\tau$. Assuming a suitably similar vector $\hat{\mathbf{r}}^\tau$ of player locations in role order exists, we define the optimal assignment of roles as the permutation matrix $\mathrm{x}_t^{\tau\star}$ which minimises the square L_2 reconstruction error

$$\mathrm{x}_t^{\tau\star} = \arg\min_{\mathrm{x}_t^\tau} \|\hat{\mathbf{r}}^\tau - \mathrm{x}_t^\tau \mathbf{p}_t^\tau\|_2^2. \tag{12.6}$$

This is the linear assignment problem where an entry $\mathrm{C}(i, j)$ in the cost matrix is the Euclidean distance between role locations

$$\mathrm{C}(i, j) = \|\hat{\mathbf{r}}^\tau(i) - \mathbf{p}_t^\tau(j)\|_2. \tag{12.7}$$

The optimal permutation matrix can be found in polynomial time using the Hungarian (or Kuhn-Munkres) algorithm [21].

12.6.3 Assignment Initialization

To solve the assignment problem, we need a reference formation to compare to. Using the mean formation (see Fig. 12.10) is a reasonable initialization as the team should maintain that basic formation in most circumstances. However, in different areas of the field there are subtle changes in formation due to the what the opposition are doing as well as the game-state. To incorporate these semantics, we used a codebook of formations which consists of every formation within our training set. However, this mapping is difficult to do as the input features have no assignment. Given we have the assignment labels of the training data, we can learn a mapping matrix $\mathbf{W}$ from the mean and covariances of the training data to its assignment labels via the linear transform $\mathbf{X} = \mathbf{W}^T\mathbf{Z}$. Given N training examples, we can learn $\mathbf{W}$ by concatenating the mean and covariance into an input vector $\mathbf{z}_n$, which corresponds to the labeled formation $\mathbf{x}_n$. We compile all these features into the matrices $\mathbf{X}$ and $\mathbf{Z}$, and given these, we use linear regression to learn $\mathbf{W}$ by solving

$$\mathbf{W} = \mathbf{X}\mathbf{Z}^T(\mathbf{Z}\mathbf{Z}^T + \lambda\mathbf{I})^{-1} \tag{12.8}$$

where λ is the regularization term. Using this approach, we can estimate a labelled formation from the training set which best describes the current unlabeled one. In terms of assignment performance on the test set, this approach works very well compared to using the mean formation for both the identity and role labels as can be seen in Table 12.5. Figure 12.12 shows the confusion matrices for both Team A and Team B for both representations. It is worth noting that the role representation gave far

Table 12.5 Accuracy of the assignment using a mean formation as well as a codebook of possible formations

	Prototype	Hit rate	
		Team A	Team B
Identity	Mean formation	38.36	29.74
	Codebook	49.10	37.15
Role	Mean formation	49.47	50.30
	Codebook	**74.18**	**69.70**

better results than the identity representation, which is not surprising seeing that only spatial location is used. In terms of the role representation (bottom two plots), it can be seen that there is little confusion between the 3 defenders (LB, CH, RB) and the 3 forwards (LW, CF, RW). However, the midfield 4 (LH, RH, IL, IR) tend to interchange position a lot causing high confusion. Noticeably, there is a discrepancy between Team A and Team B which is understandable in this case as Team B interchanges positions more than twice the amount than Team A upon analysis of the ground-truth.

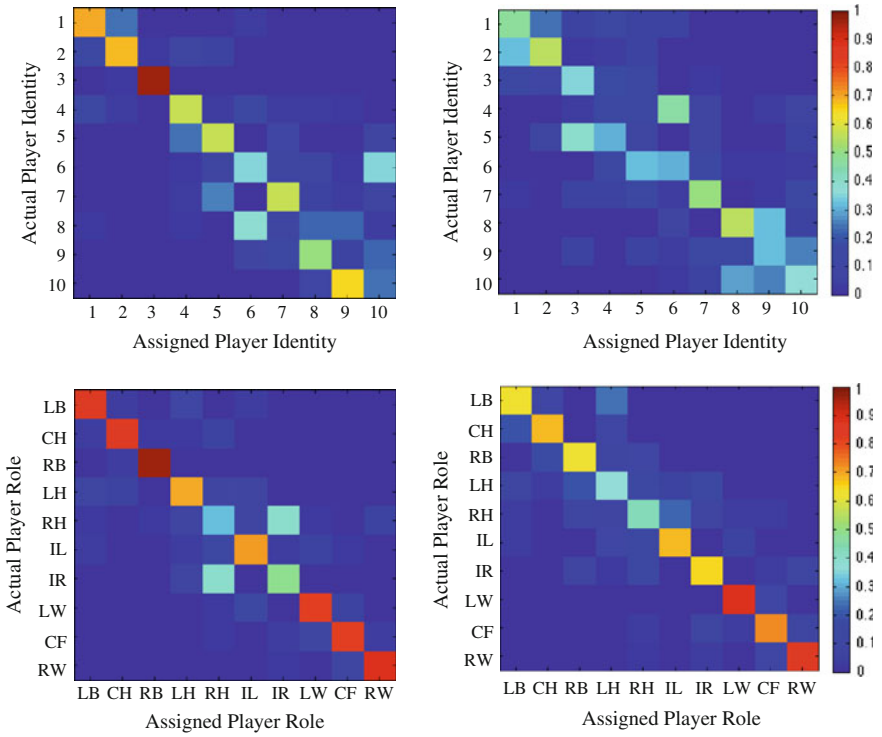

Fig. 12.12 Confusion matrix showing the hit-rates for correctly assigning identity (*top row*) and role (*bottom*) for Team1 (*left*) and Team 2 (*right*) on the test set

12.7 Interpreting Noisy Data

In practice, we will not obtain perfect data from a vision-system so our method has to be robust against both missed and false detections. Given the four annotated matches (presented in Table 12.3), the precision and recall rates for the detector and the team affiliation are given in the left side of Table 12.6. In this work, we consider a detection to be made if a player was within 2 m of a ground-truth label.

12.7.1 Assigning Noisy Detections

To determine whether or not we should make the assignment or discard the detection, some type of game context feature is required (e.g. the part of the field most of the players are located). To do this, we employed a similar strategy to the one we proposed in Sect. 12.6.3. However, instead of learning the mapping from the clean features $\mathbf{Z}$, we learn from the noisy features $\mathbf{Z}_{\text{noisy}}$. As the player detector has systematic errors (there are some "black-spots" on the field due to reduced camera coverage, or game situations where players bunch together), we include the number of players detected from the system as well as the mean and covariance in our noisy game context feature $\mathbf{z}_{\text{noisy}}$, which we can then use to learn $\mathbf{W}_{\text{noisy}}$. We are able to do this as we make the assumption that the clean centroid is a good approximation to the noisy centroid which was found to be a valid assumption as can be seen in Fig. 12.13. Using this assumption, we can obtain a reasonable prototypical formation to make our player assignments.

Using the estimated prototype, we then make the role assignments using the Hungarian algorithm. This is challenging however, as we may have missed or false detections which alters the one-to-one mapping between the prototype and input detections. To counter this, we employed an "exhaustive" approach, where if we have fewer detections than the number of players in the prototype, we find all the possible combinations that the labels could be assigned then use the combination which yielded the lowest cost from the assignments made. Conversely, if we had more detections than the number of players, we find all the possible combinations that the detections could be and then use the combination of detections which had the lowest cost.

For example, given we have only 9 detections for a team, we first find the 10 possible combinations that prototype could be (i.e. $[1, \ldots, 9]$, $[2, \ldots, 10]$,

Table 12.6 Precision-Recall rates for the raw detections (left) and with the initialised assignments (right)

	Raw detections		With assignment	
	Precision	Recall	Precision	Recall
Detections	77.49	89.86	**91.90**	80.46
Team A	72.54	86.14	**86.69**	74.17
Team B	79.84	89.66	**92.91**	82.85

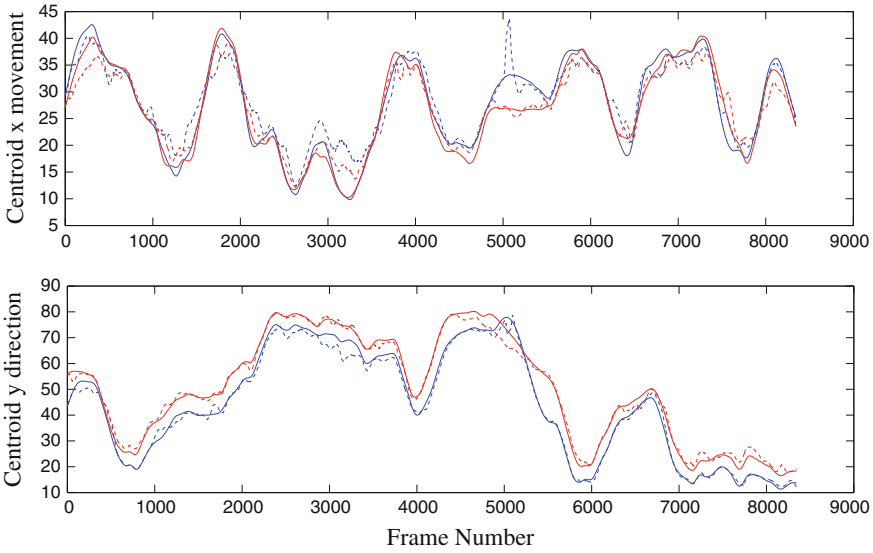

Fig. 12.13 As the centroids of both the clean (*solid*) and noisy (*dashed*) of both teams ($blue$ = team1, red = team2) are roughly equivalent, we learn a mapping matrix using linear regression to find a formation from the training set which can best describe the noisy test formation

[1, 2, 4, . . . , 10], [1, 2, 3, 5, . . . , 10] etc.). For each one of these combinations, we then perform the Hungarian algorithm and calculate the cost of the made assignments. After we have exhaustively gone through all possible combinations, we make the assignment based on the combination with the lowest cost. Or given we have 11 detections for a team, we first find the 11 possible combinations that the detections could be, find the cost for each set and choose the one with the lowest cost. However, sometimes we get false positives which means that even though we may get 10 detections for a team we may only have 7 or 8 valid candidates. Employing this approach greatly improves the precision rate, while the recall rate decreases which is to be expected (see right side of Table 12.6). Even despite the drop in recall, we still assign role reasonably well (over 55 % compared to 66 % on the clean data) as can be seen in Table 12.7.

Table 12.7 Detection rates assigning roles to the noisy data. The column on the far right gives the effective hit-rate (i.e. missed detections omitted) of the correct assignments

	Correct	Incorrect	Missed	Hit rate
Team A	41.89	32.89	25.22	56.02
Team B	45.92	35.56	18.53	56.36

12.7.2 Denoising the Detections

While our precision and recall rates from the detector are relatively high, to do useful analysis we need a continuous estimate of the player label at each time step to do formation and play analysis. This means that we need a method which can de-noise the signal—i.e. a method which can impute missing data and filter out false detections. Given the spatial bases, the bilinear coefficients and an initial estimate of the player labels, we can use an Expectation Maximization (EM) algorithm to denoise the detections. The approach we use is similar to [3]. Using this approach, the expectation step is simplified to making an initial hard assignment of the labels which can be gained by finding the initial assignments using the method described in the previous section. From this initialization, we have an initial guess of $\hat{\mathbf{S}}$. In the maximization step, we can calculate $\mathbf{C} = \Theta^T \hat{\mathbf{S}}\mathbf{B}$, and then estimate $\mathbf{S}$ from our new $\mathbf{C}$ as well as our spatial and temporal basis $\mathbf{B}$ and Θ. An example of the cleaned up detections using this approach is shown in Fig. 12.14.

As the recall rate of the denoised data is 100 %, we are interested to see how precise our method is in inferring player position based on their label. To test this, we calculated the precision rate for the detections and the denoised detections against a distance threshold—that is, the minimum distance a player had to be to ground-truth to be recognised as a correct detection). The results are shown in Fig. 12.15. As can be seen from these figures, the detections from the player detector are very accurate and do not vary with respect to the error threshold (i.e. it either detects a player very precisely or not at all). Conversely, the denoised data is heavily smoothed due to the bilinear model, so we lose some of the finer detail to gain a continuous signal.

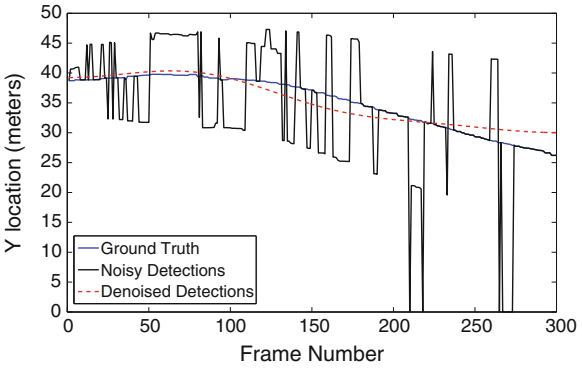

Fig. 12.14 Given our noisy detections (*black*), using our bilinear model we can estimate the trajectory of each player over time. We can see our estimate (*red*) is close to the ground-truth (*blue*)

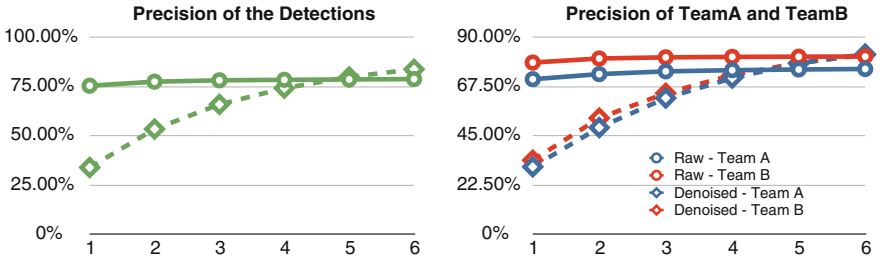

Fig. 12.15 Precision accuracy versus the distance threshold from ground truth (in metres) (*left*) the overall detections, (*right*) the detections based on team affiliation. The *solid lines* refer to the raw detections and the *dashed lines* refer to the denoised signal

12.8 Summary

Accurately tracking players over long durations of time is an unsolved computer vision problem, and prevents automated analysis of team sports using traditional representations based on player tracks. In this paper, we instead directly modelled team behaviours from raw player detections, and compared two representations which are robust to missed and false detections: (1) occupancy maps and (2) a bilinear-spatiotemporal basis model. We demonstrated that occupancy map features can accurately represent global group behaviours, and can be used to recognise group activities corresponding to important game states from raw player detections, without player tracking or ball information. However, one of the challenges of an occupancy map representation is that high dimensionality is required, and it is difficult to model team behaviours at the level of individual players. To overcome this, we proposed the use of a bilinear spatiotemporal basis model using a *role representation* to clean-up the noisy detections which operates in a low-dimensional space. This provides a very compact representation of group behaviours, and can facilitate tasks such as clustering and retrieval. We evaluated our approach on approximately 200,000 frames of field-hockey data from a state-of-the-art real-time player detector.

Acknowledgments The QUT portion of this research was supported by the Queensland Government's Department of Employment, Economic Development and Innovation.

References

1. Akhter I, Sheikh Y, Khan S, Kanade T (2008) Nonrigid structure from motion in trajectory space. In: NIPS
2. Akhter I, Sheikh Y, Khan S, Kanade T (2010) Trajectory space: a dual representation for nonrigid structure from motion. T. PAMI
3. Akhter I, Simon T, Khan S, Matthews I, Sheikh Y (2012) Bilinear spatiotemporal basis models. ACM Trans Graph
4. Arikan O (2006) Compression of motion capture databases. ACM Trans Graph 25(3)

5. Avrahami-Zilberbrand D, Banerjee B, Kraemer L, Lyle J (2010) Multi-agent plan recognition: formalization and algorithms. In: AAAI
6. Beetz M, von Hoyningen-Huene N, Kirchlechner B, Gedikli S, Siles F, Durus M, Lames M (2009) ASPOGAMO: automated sports game analysis models. Int J Comput Sci Sport 8(1)
7. Bregler C, Hertzmann A, Biermann H (2000) Recovering non-rigid 3D shape from image streams. In: CVPR
8. Bronstein A, Bronstein M, Kimmel R (2008) Numerical geometry of non-rigid shapes. Springer, Berlin
9. Carr P, Sheikh Y, Matthews I (2012) Monocular object detection using 3d geometric primitives. In: ECCV. Springer
10. Chang M, Krahnstoever N, Ge W (2011) Probabilistic group-level motion analysis and scenario recognition. In: ICCV
11. Cootes T, Taylor C, Cooper D, Graham J (1995) Active shape models—their training and applications. Comput Vis Image Underst 61(1):38–59
12. D'Orazio T, Leo M (2010) A review of vision-based systems for Soccer video analysis. Pattern Recognit 43(8)
13. Gupta A, Srinivasan P, Shi J, Davis L (2009) Understanding videos, constructing plots: learning a visually grounded storyline model from annotated videos. In: CVPR
14. Hervieu A, Bouthemy P (2010) Understanding sports video using players trajectories. In: Zhang J, Shao L, Zhang L, Jones G (eds) Intelligent video event analysis and understanding. Springer, Berlin
15. Hess R, Fern A (2009) Discriminatively trained particle filters for complex multi-object tracking. In: CVPR
16. Hess R, Fern A, Mortensen E (2007) Mixture-of-parts pictorial structures for objects with variable part sets. In: ICCV
17. Huang C, Shih H, Chao C (2006) Semantic analysis of soccer video using dynamic bayesian networks. T. Multimed 8(4)
18. Intille S, Bobick A (1999) A framework for recognizing multi-agent action from visual evidence. In: AAAI
19. Intille S, Bobick A (2001) Recognizing planned, multi-person action. Comput Vis Image Underst 81:414–445
20. Kim K, Grundmann M, Shamir A, Matthews I, Hodgins J, Essa I (2010) Motion fields to predict play evolution in dynamic sports scenes. In: CVPR
21. Kuhn HW (1955) The Hungarian method for the assignment problem. In: Naval research logistics quarterly
22. Lazarescu M, Venkatesh S (2003) Using camera motion to identify different types of American football plays. In: ICME
23. Li R, Chellappa R (2010) Group motion segmentation using a spatio-temporal driving force model. In: CVPR
24. Li R, Chellappa R, Zhou S (2009) Learning multi-modal densities on discriminative temporal interaction manifold for group activity recognition. In: CVPR
25. Liu T, Ma W, Zhang H (2005) Effective feature extraction for play detection in American football video. In: MMM
26. Money A, Agius H (2008) Video summarisation: a conceptual framework and survey of the state of the art. J Vis Commun Image Represent 19(2):121–143
27. Morariu V, Davis L (2011) Multi-agent event recognition in structured scenarios. In: CVPR
28. Perse M, Kristan M, Kovacic S, Pers J (2008) A trajectory-based analysis of coordinated team activity in basketball game. Comput Vis Image Underst
29. Rao K, Yip P (1990) Discrete cosine transform: algorithms, advantages, applications. Academic, New York
30. Sadilek A, Kautz H (2008) Recognizing multi-agent activities from GPS data. In: AAAI
31. Siddiquie B, Yacoob Y, Davis L (2009) Recognizing plays in American football videos. Technical report University of Maryland

32. Stracuzzi D, Fern A, Ali K, Hess R, Pinto J, Li N, Konik T, Shapiro D (2011) An application of transfer to American football: from observation of raw video to control in a simulated environment. AI Mag 32(2)
33. Sukthankar G, Sycara K (2008) Hypothesis pruning and ranking for large plan recognition problems. In: AAAI
34. Sukthankar G, Sycara K (2012) Activity recognition for dynamic multi-agent teams. ACM Trans Intell Syst Technol
35. Torresani L, Bregler C (2002) Space-time tracking. In: CVPR
36. Xu C, Zhang Y, Zhu G, Rui Y, Lu H, Huang Q (2008) Using webcast text for semantic event detection in broadcast. T. Multimed 10(7)
37. Zhang Y, Ge W, Chang M, Liu X (2012) Group context learning for event recognition. In: WACV