# A Trajectory-Based Analysis of Coordinated Team Activity in Basketball Game

Matej Perše, Matej Kristan, Stanislav Kovačič, Janez Perš

*Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, SI-1000 Ljubljana, Slovenia*

**Published in: Computer Vision and Image Understanding**
**doi:10.1016/j.cviu.2008.03.001**

**Accepted: 9 March 2008**

**Abstract**

This paper proposes a novel trajectory-based approach towards automatic recognition of complex multi-player behavior in a basketball game. First, a probabilistic play model is used on player trajectory data to segment the play into game phases (offense, defense, time-out). This way, both the temporal boundaries of the observed activity and its broader context are obtained. Next, the team activity is analyzed in more detail by detecting key elements of the basketball play. Following the basketball theory, these key elements (starting formation, screen, move) are the building blocks of basketball play, and therefore, their temporal order is used to produce the semantic description of the observed activity. Finally, the recognition of the activity is achieved by comparing its semantic description to the descriptions of manually defined templates, stored in the database. The effectiveness and robustness of the proposed approach is demonstrated on 71 examples of three types of basketball offense.

*Key words:* Complex behavior, Behavior recognition, Play segmentation, Semantic description, Basketball

## 1 Introduction

One of the paramount problems in sport science is objective analysis of player performance. While individual player's physical abilities can be readily tested in laboratory conditions, the team performance can only be observed during the actual gameplay. This process may include advanced analysis methods, such as video recording and statistical analysis, but it nevertheless relies on observation and manual annotation by sport experts, with the potential risk of becoming too subjective. Additionally, manual annotation is a time consuming and tedious task, mostly limited either to the academic research or to the small number of teams which can afford the sufficient number of qualified experts. Moreover, some researchers have found [1] that even sport experts often cannot observe and recall all the details which can prove crucial for the correct interpretation of results.

Therefore, an increasing volume of research concerned with automatic or semi-automatic recognition and analysis of human behavior in sports is not surprising. The ultimate goal of such research is to develop methods for automatic interpretation and analysis of team performance, which would present a concise summary of the team's and players' strengths, weaknesses and mistakes. In addition to that, the same methods could be used in many other areas, such as sports broadcasting, as a tool for automatic extraction of the course of the gameplay, either for the purpose of enhancing live broadcasts or facilitating easier video archival. In the broader context, similar methodology could be used in human motion analysis for video surveillance, ambient assisted living, and similar tasks. However, the main focus of this article is the challenge of observing the sports match and interpreting the team activity on the field.

The quality of the team has two important components. The first component encompasses the skills of the players, and is expressed as their technical knowledge. The second component is expressed as the overall *team tactics*. In order for a team to be successful, it needs individuals with excellent technical skills. Nevertheless, these individuals have to be able to act together as a group - a task that requires good coordination between individual players and can be achieved only by significant amount of training. Following this challenge, the focus of this article is on the analysis of coordinated activity in team sports, in particular basketball.

It is widely accepted [2] that the most successful players have a capability to react differently in similar situations. On the team level, the situation is similar - good teams can quickly change their tactics if needed. Such behavior prevents the opposing teams to prepare a good defense, and keeps the play interesting, as both teams have to continuously adapt to the situation on the court. This introduces a certain level of complexity and randomness in the team performance, and makes the design of entirely automatic analysis system, which would recognize, understand and grade every possible situation on the court extremely difficult. However, due the nature of sport rules and rigorous player training, the motion of players across the field is not entirely random, and it is reasonable to expect that it is possible to extract some common features of the team, especially when considering that a coordinated team play is usually trained in advance.

The remainder of this paper is structured as follows. In the rest of this section we give a short overview of the related work and present the concept of our approach. The methods for segmentation and recognition of the complex multi-agent behavior are presented in Section 2, and in Section 3 the experimental results are presented. Finally, the summary and future work are given in Section 4.

### *1.1 Related work*

Several different approaches to the problem of motion analysis and activity recognition have been proposed. They could be divided into two categories based on the type of data they work on:

- the analysis of single or multi-person activity from raw visual data, and
- the analysis of single or multi-person activity from temporal trajectories.

There are alternative classifications of the motion analysis approaches [3, 4], however, this one was chosen due to its generality and suitability to classify the work, related to the sport domain. Although our approach works on trajectory data, we also present the short overview of the sports-related research based on the raw visual data. We deem this appropriate, since the general analysis concepts of the two categories coincide in several aspects.

#### *1.1.1 The analysis of activity from raw visual data*

This category was extensively covered in the survey of Gavrila [3], where several approaches to the "Looking at People" domain are discussed, with the emphasis on the visual analysis of gestures and whole body motion. Analysis approaches are divided into three different categories according to the type of the model used to represent the motion - the 2D approaches with or without explicit shape models and the 3D approaches. As the author states, these approaches can be found in a number of promising applications, such as virtual reality, "smart" surveillance systems, advanced user interfaces, model-based video encoding and motion analysis. In recent years many interesting sport-related applications appeared in the motion analysis and modeling domain, such as content-based indexing of video footage [5, 6], or learning motion models in golf [7], soccer [8], baseball [9], or choreography in dance [10] and acrobatics [11].

The emphasis of the work from this category is generally focused on matching an unknown test sequence with a library of labeled training sequences, which are learned from training examples [3]. Different techniques are used to describe complex dynamics of the continuous processes that can be observed from the human behavior such as using the spatio-temporal templates to represent the walk [12], using phase space constraints to represent the body motion [13] or the use of Hidden Markov Models (HMMs) to represent states of the visual behavior and the transitions between them [14]. These techniques can be successfully implemented in sport and can provide athletes the feedback needed to help them improving their kinematic skills. Regarding the applications, the emphasis of these approaches is usually on teaching and monitoring the correct execution of certain predefined motions, such as ball handling and throwing in basketball, or for example swinging of a golf club [7].

#### *1.1.2 The analysis of single and multi-person activity from temporal trajectories*

The techniques described so far usually do not explicitly consider the global spatial properties of the observed behavior, which may be of significant importance. Let us for example imagine a basketball player, making a shot. By using the techniques described in previous section, we may learn that a shot was executed perfectly. However, a larger context of this activity has to be obtained in some other way. For example, we would like to know if the situation on the court warranted execution of the shot at this particular time and place. To answer this question, we would like to know if any of player's teammates were in an even better position for the shot, and whether the opponents were in such formation that his shot was really a good choice. To answer these questions,we need to turn to the analysis of trajectories, obtained by tracking of individual players.

The methods of trajectory-based activity analysis can be divided into the two groups. The first group represents the methods which deal with modeling of the statistical distributions of trajectories. Johnson and Hogg [15, 16] presented two approaches for modeling the variable non-linear behaviors. In the first approach, a competitive learning neural network was used on flow vectors from the image sequences of pedestrians [15], and in the other, the probabilistic motion model was obtained using Gaussian mixture model, representing the system state changes of the pedestrian motion [16].

The second group of trajectory-based analysis of human behavior involves the more sophisticated modeling of coordinated group activity, which usually involves some additional information provided by the domain expert [17, 18, 19]. Li and Woodham [19] present a concept system to represent and reason about selected hockey plays based on the trajectory data, augmented with domain knowledge such as forward/backward skating, puck possession, etc. The Finite State Machine (FSM) model is used as a mechanism for representing, identifying the observed activity and reasoning about possible better outcomes of the observed situation. Intille and Bobick [17] have built models of the football plays using belief networks and temporal graphs. A similar approach was used by Jug et.al [18] to assess the team performance in basketball offense. The main contribution of the latter two approaches is the representation of multi-agent activity and recognition from noisy trajectory data. This is done by dividing the multi-agent activity into the individual visually-grounded, goal-based primitives which are probabilistically integrated by the low-order temporal and logical relationships. However, there are two main problems with such approach. The first one is the need for precise temporal segmentation of the analyzed trajectories. The second problem is the difficulty of building temporal and logical relationships, especially due to many different parameters which need to be defined manually. Therefore, such approach is not particularly suited in cases when either large quantity of data has to be analyzed, or many different behavior models are used in the analysis.

### *1.2 Our approach*

In our work, we address the problem of trajectory-based analysis of basketball game, with the goal of overcoming the described problems. We chose our approach with the expert sport knowledge in mind. Similarly to the procedure used in sport research, we perform a two step analysis process, where the match is first segmented according to phases of the play (offense, defense, time-out), and then the detailed analysis of each segment is performed.

In the first step, a Gaussian mixture model is used to segment the continuous player trajectories into shorter game segments, corresponding to offense, defense and timeouts. This stage provides us both with rough segment boundaries and with the broader context of the game inside a particular segment. In the second step, we are able to perform the recognition of particular type of basketball activities. In this article, we only focus on the recognition of the organized activity in the offensive part of the game. The basketball offense is the most interesting and widely studied [2, 20] part of the game for coaches and basketball experts. It is trained in advance and the set of trained offenses for a particular team does not change significantly during the course of one match.

To perform the recognition of activity, we developed three different trajectory-based detectors of key basketball elements, which are the theoretical building blocks of any basketball play. These detectors are used to transform player trajectories, which represent observed activity, into the sequence of symbols - the semantic description. Obtained symbols describe the actions and interactions between players, and contain the notation of the observed element and its observed position on the court. To determine the similarity between the observed activity and the predefined activity template, we simply compute a Levenstein distance between the sequence of symbols obtained from the trajectories of the players and the sequence of symbols obtained from the activity template. By repeating the template-trajectory matching procedure for every template available, we can find the template with the shortest distance to the sequence of symbols, obtained from trajectory data. If the distance is below a certain threshold, we assign the label of the most similar template to the observed trajectory segment.

Beside removing the need for manual segmentation, an important contribution of our approach is the simplified process of providing expert knowledge in machine-suitable format. The method used by Intille and Bobick [17] consists of providing expert knowledge in the form of temporal and logical relationships in belief network, for every activity that has to be recognized. This process is unsuitable for field experts (i.e. sport coaches), slow, and can be inaccurate or subjective because of non-obvious relationships between the activity and the network structure [17, 18]. In our case, the specific structure of basketball activity is encoded in the form of activity templates, which can be represented graphically and are significantly more familiar to the average sport expert.

## 2 Methods

This section describes the methods, developed for analyzing player motion data in the context of cooperative basketball play. We assume that we have trajectory data of all players available for the whole duration of the match. In Section 3, we will show that given current state of the technology, this is not unreasonable.

The analysis is conducted in two steps. First, the players' trajectories are temporally segmented into three game phases - offense, defense and time out. The segmentation is achieved using the probabilistic game model, presented in Section 2.1. Next, the template-based recognition procedure is applied to every individual segment of the match (Section 2.2).

Team sports are determined both by their rules and by the collective goals, which the teams must pursue to defeat their opponents. In many popular team sports, the court is divided into two halves, and the teams' activities alternate between offense and defense, with minor interruptions, such as time-outs, free throws and free kicks. Therefore, the game could be regarded as a process, consisting of certain number of discrete phases. These phases correspond to offensive play, defensive play, time outs, inactive play, free throws, free kicks, and other miscellaneous activities.

In case of basketball, our model assumes that the play contains the following three phases: offensive play ($m_1$), defensive play ($m_2$) and time outs ($m_3$):

$$M = \{m_1, m_2, m_3\} . \tag{1}$$

Our basic assumption is that there exists intrinsic relation between parameters of player motion (position, velocity and direction) and game phases. Therefore, our model is based solely on the observation of players' motion. Similar to Erdmann [21], we calculate the *collective* position of players by calculating the team gravity center, (i.e. the mean position of $N$ players, belonging to single team), and observe the two dimensional motion of this single point across the court. Therefore we define the *flow vector* $\mathbf{x}(\mathrm{t})$ as

$$\mathbf{x}(t) = [x_t, y_t, \Delta x_t, \Delta y_t]^T \ , \tag{2}$$

where $x_t$ and $y_t$ represent the position of the team gravity center

$$x_t = \frac{1}{N}\sum_{j=1}^{N} x_j, \qquad y_t = \frac{1}{N}\sum_{j=1}^{N} y_j, \tag{3}$$

and $\Delta x_t$ and $\Delta y_t$ represent the corresponding velocity components of the gravity center at time $t$

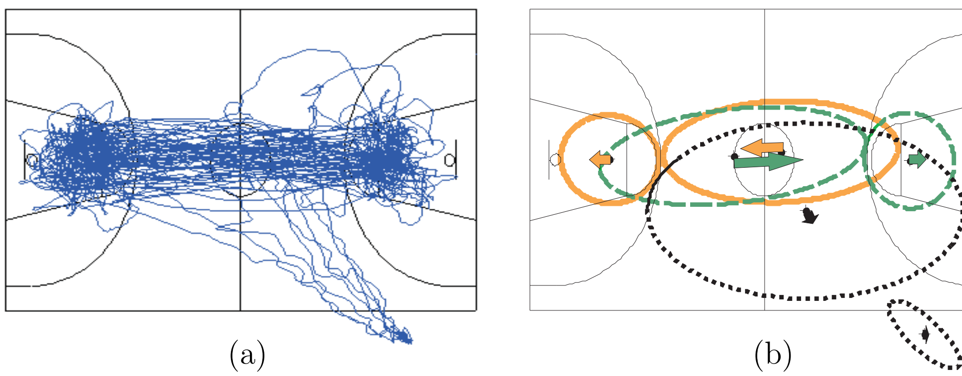
$$\Delta x_t = x_t - x_{t-1}, \qquad \Delta y_t = y_t - y_{t-1}. \tag{4}$$

To account for the variability and uncertainty in player motion and team behavior, we define a probabilistic model of the game phases using a mixture of Gaussians [22]

$$p(\mathbf{x}|m_i) = \sum_{k=1}^{n} \alpha_k^{(i)} \cdot p(\mathbf{x}|\mu_k^{(i)}, \Sigma_k^{(i)}), \qquad (i = 1, 2, 3), \tag{5}$$

where $m_i$ represents game phase, parameters $\alpha_k$ represent the mixing coefficients such that $\sum_{k=1}^{n} \alpha_k^{(i)} = 1$, $p(\mathbf{x}|\mu_k^{(i)}, \Sigma_k^{(i)})$ is the $k$-th Gaussian density function with mean $\mu_k^{(i)}$ and the covariance matrix $\Sigma_k^{(i)}$, and $n$ represents the number of Gaussian density functions used to model each game phase.

To determine the parameters of the Gaussian density functions, we use the Expectation Maximization (EM) algorithm [23] on a manually labeled training sequence. As an illustration, Figure 1 shows the obtained game model, which uses two components ($n$=2) per each phase. It can be observed that similar, but mirrored likelihood functions are obtained for the phases which represent offensive and defensive play.



(a) (b)

Fig. 1. (a) Trajectory of the team gravity center. (b) Gaussian mixture models for the three game phase phases obtained by manually labeling the trajectory from figure (a); ($p(\mathbf{x}|m_1)$ - orange full line, $p(\mathbf{x}|m_2)$ - green dashed line, $p(\mathbf{x}|m_3)$ - black dotted line, where $m_1, m_2$ and $m_3$ denote the offensive play, defensive play and time outs). The arrows show the direction and magnitude of velocity component of the flow vector.

Once the game model is built, we can calculate the probability of model $m_i$ given current *flow vector* $\mathbf{x}$(t) using Bayes formula

$$p(m_i|\mathbf{x}(t)) = \frac{p(\mathbf{x}(t)|m_i)p(m_i)}{p(\mathbf{x}(t))}, \qquad (i = 1, 2, 3), \tag{6}$$

and given that the probability $p(\mathbf{x}(t))$ remains constant for all models $m_i$, the classification of the given sample $\mathbf{x}(t)$ at time $t$ is expressed as

$$m_*(t) = \arg\max_{m_i \in \mathrm{M}} \{p(\mathbf{x}(t)|m_i)p(m_i)\}, \qquad (i = 1, 2, 3), \tag{7}$$

where $p(m_i)$ is the *a priori* probability of phase $m_i$. These probabilities were estimated in advance by roughly estimating the amounts of time teams usually spend in each individual game phase.

$$p(M) = [p(m_1), p(m_2), p(m_3)] = [0.45, 0.45, 0.10] \,. \tag{8}$$

This method provides reasonably good classification for the individual time instants. However, when used for trajectory segmentation, it produces a number of extremely short segments, as it does not enforce any temporal continuity. Thus, we enforce this requirement by smoothing the output of the classification process using the nonlinear kernel in the following form:

$$m_{**}(t) = \arg\max_{m_i \in \mathrm{M}} \left\{ \sum_{k=t-K}^{t+K} D_{m_*,m_i}(k) \right\}, \quad D_{m_*,m_i}(k) = \left\{ \begin{array}{ll} 1; & m_*(k) = m_i \\ 0; & otherwise \end{array} \right\} . \tag{9}$$

This way, the $t$-th sample is assigned the label that receives the highest score among the observed individual labels inside the kernel window of length $2K+1$. The kernel width is set to twice the length of theoretically shortest possible segment of play, which in basketball corresponds to approximately three to four seconds.

### *2.2 Recognition of complex play activities*

The previously described segmentation method provides us with the short segments of the match, labeled as "offense", "defense" or "time-out". This information puts each of the segments in its appropriate context, which in turn enables to handle different contexts of the play in different ways. In this paper, we focus only on the analysis of basketball offense.

Once the segmentation is performed, we know that the activity of interest starts and ends somewhere inside the segment. The information we don't have and wish to obtain is, which of the known activities (known types of basketball offense) is the team performing. For that purpose, we represent the team activity as a sequence of symbols, which describe players' actions with sufficient degree of detail.

#### *2.2.1 Representation of complex play activities and activity templates*

In sport theory, a play is comprised of basic *(key)* elements of the play. These elements are sport specific; they may vary significantly from sport to sport and depend heavily on the rules of a particular sport. In basketball and European handball, for example, a common element of play is *screening*, where a player tries to make the space for his teammate by blocking the teammate's defender.

This way, the teammate gets a better chance of scoring. Another element, called *cutting*, can be observed in basketball or soccer, where player tries to cut his way into the empty space on the court, where he has a better chance of receiving the ball and scoring.

We designed the method for play recognition, where we rely on detection of three basic elements of the basketball play - *screen*, *move* and *player formation*. These elements form a dictionary, which is, along with their spatial and temporal relations, used to build semantic descriptions of player and team activities, in our case, basketball offenses. Such approach transforms a group of player trajectories, obtained by tracking, into a stream of corresponding symbols. These symbols can also be, if needed, represented in a human readable form. They represent a simple narrative of what was the team doing on the court. Thus, a complex problem of multiple agents performing a complex activity is transformed into a problem of analyzing sequences of symbols, which represent a course of a play.

The recognition is achieved by comparing the semantic description of observed trajectory segment with semantic descriptions, generated from the activities, stored in the activity database.

These stored activities are called *activity templates*. They are detailed, machine readable representations of any basketball activity, and can be automatically rendered into semantic descriptions when needed. They contain the sequence of key elements, and their (ideal) positions on the court in form of absolute court coordinates. Activity templates also contain associations between key elements of the play and the players, which should perform them. Players are denoted only by abstract indexes (numbered from 1 to 5, since there are 5 players in the basketball team). Activity templates may contain activity in greater detail than the resulting semantic descriptions and can be rendered to their graphical representation as well, to help users who design them.

Using this representation of basketball play, our problem of recognition and analysis of complex activities has been transformed into the problem of comparing two sequences of symbols, where one sequence belongs to the trajectory segment being analyzed, and the other belongs to one of the plays from database. The calculated distance between the two sequences can be used in two ways. First, it enables us to find the most similar activity from the database of known activities. Second, it can be interpreted as a measure of quality of team performance - if the distances to all of the stored plays are large, the activity is either not in database, or, it was just performed poorly. In case that the activity is in the database, the distance increases with the increasing discrepancies between the observed activity and the activity from the database, which may indicate poorly trained team or the team which has trouble with opponent's defense.

In order to derive semantic descriptions from trajectory data, we had to model the expert knowledge, used in the basketball community. Besides the activity representation itself, this includes modeling of the court, which is used to determine the spatial characteristics of the observed elements, and modeling of the key elements. The derivation of this knowledge is described next.

### *2.2.2 Court partitioning*

In the process of transforming trajectory data into sequence of symbols, we need to preserve certain amount of spatial information. We encode spatial position of screens and player moves according to the region of the court where these elements have been observed. Thus, a certain degree of generality is provided, as the elements, observed in same *regions*, produce the same spatial encoding, regardless of the actual position of the element within the region. This way, for example, a player movement across the court is encoded as the sequence of crossing region boundaries, and we can write it in human readable form as

M_PlayerX_CourtRegionA_CourtRegionB,

which denotes the *movement* (M) of player X from the court region A to the court region B. Similarly, the notation of the *screen* (SCR), which involves the actions of two players (X and Y) in some court region A can be written as:

SCR_PlayerX_PlayerY_CourtRegionA.

Partitioning of the court could be done in many different ways, preferably based on expert knowledge. Regrettably, after a brief study of relevant basketball literature and after consultations with several basketball experts, we did not come across any consistent solution of this problem. The main reason appears to be the fact that coaches usually think of the game in the form of playing roles (e.g. *point guard, forward or center*), which, in general, do not carry sufficient information to provide universal partitioning of the court. Therefore, we developed our own spatial court model.

To do this, we assembled a set of 34 manually defined activity templates, which corresponded to 17 different types of offensive basketball play. Each type of basketball offense was represented by the left and right version of the template, since most of the plays can be performed either in original variation, or mirrored across the larger of the court axes. The plays were taken from appropriate basketball literature [2, 20]. We extracted absolute positions of *key elements* from those templates.

The positions of all key elements from those templates were used to divide the court into different numbers of non-overlapping regions. To obtain region centers and their boundaries, we applied the k-means clustering algorithm [24] to the complete set of element positions. For comparison, we also built a "naive" court model, which was obtained by dividing the court into the same number of equally sized rectangular regions. This way, we built a collection of spatial court models, which were evaluated in our wider play analysis framework, regarding their influence on the overall recognition rate. For illustration, Figure 2 shows the partition, which yields the best recognition results on our test data, and was obtained using the k-means algorithm. Names of the regions have been chosen according to the player roles, which are commonly associated with corresponding areas of the court.

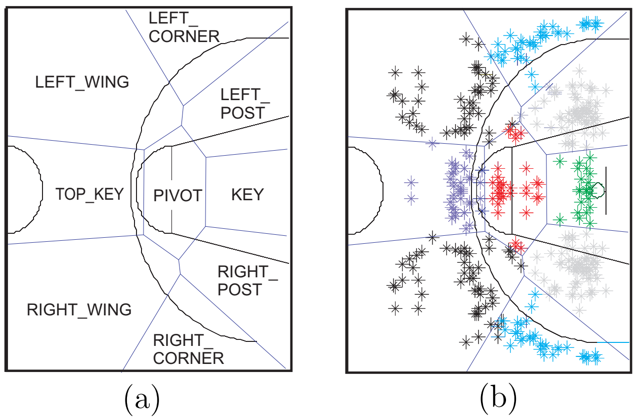



Fig. 2. (a) Names of the regions, chosen with regard to the player roles. (b) Asterisks show positions of elements (*screens, moves*) extracted from the 34 activity templates and used in the process of obtaining spatial court model.

### *2.2.3 Detection of key elements*

We decided to observe three key elements of the basketball game: *starting formation* of the team, player motion around the court (*moves*) and *screens*. All of these elements can be observed from trajectory data without any additional annotations using the set of methods we call *key element detectors*.

***Starting formation*** The starting formation is detected by comparing a set of the current positions of the players $X_t = \{\mathbf{x}_t^{(n)}\}_{n=1}^N$ to a set of reference starting positions $X_{ref} = \{\mathbf{x}_{ref}^{(n)}\}_{n=1}^N$ which are provided by each activity template. N denotes number of the players in one team. The closer the current positions are to the reference positions, the likelier it is that the formation took place. However, those positions, which are closer to the basket usually bear more weight in determining the formation than those farther from the basket. The reasoning behind this is twofold. First, the players that are positioned closer to the basket have less maneuver space, since they can be easily blocked by the opposite team. Second, there are more court markings that can be used for precise orientation close to the basket than farther away.

Let $d_n = \|\mathbf{x}_t^{(n)} - \mathbf{x}_{ref}^{(n)}\|$ be the $l_2$ distance from the $n$-th player to the $n$-th reference position in the $k$-th formation and let $r_n = \|\mathbf{x}_{ref}^{(n)} - \mathbf{x}_{basket}\|$ be the $l_2$ distance from that reference position to the basket. The likelihood of $k$-th formation given the current set of positions $X_t$ is then defined as

$$\mathcal{L}(formation = k|X_t) \triangleq \sum_{n=1}^{N} \mathcal{N}(d_n; 0, \sigma_d) \cdot w_n, \tag{10}$$

where $\mathcal{N}(\cdot; 0, \sigma)$ denotes a zero-mean Gaussian with variance $\sigma^2$ and

$$w_n = \frac{\mathcal{N}(r_n; 0, \sigma_r)}{\sum_{n=1}^{N} \mathcal{N}(r_n; 0, \sigma_r)} \tag{11}$$

is the importance value, which determines, how much $n$-th reference position contributes to the $k$-th formation. The reasoning behind the importance weights (11) is as described above: accurate positioning on the reference positions that are closer to the basket is more important than the positioning on those farther from the basket. We define the *goodness* of the $k$-th formation by the likelihood ratio

$$S(formation = k) \triangleq \frac{\mathcal{L}(formation = k|X_t)}{\mathcal{L}(formation = k|X_{ref})}, \tag{12}$$

where $\mathcal{L}(formation = k|X_t)$ is the likelihood of the $k$-th formation given the current positions $X_t$, and $\mathcal{L}(formation = k|X_{ref})$ is the likelihood of the $k$-th formation given the reference positions $X_{ref}$ for that formation.

The parameter $\sigma_d$ from (10) determines the notion of proximity, i.e. it determines when a player is considered close to a certain reference position. In our implementation we set this parameter to $\sigma_d$=1 meter. The parameter $\sigma_r$ from (11) reflects the spatial importance which is assigned to the reference positions in a given formation with respect to their distance from the basket. In our implementation, this parameter was set to $\sigma_r$ =6.25 meters, which represents the radius of the three points-area circle on the basketball court.

The *goodness* function (12) is bounded to the interval [0..1], yielding zero at *total mismatch* and one at *perfect match*. In our experience, the set of current positions $X_t$ may be regarded as the $k$-th formation when $S(formation = k)$ exceeds the value of 0.85.

***Screen.*** In the basketball literature, the screen is defined as a close contact between two players [2], where ideally one player is standing still and the other runs in his proximity. Thus a certain interaction among two players is more likely to be interpreted as a screen if the velocity of the slower player is low and the distance between the players is small.

Let $d_t$ be the $l_2$ distance between the two interacting players and let $v_t$ be the velocity of the slower of the players. The likelihood function of a screen is defined as

$$\mathcal{L}(screen|d_t, v_t) \triangleq \mathcal{N}(d_t; 0, \sigma_d) \cdot \mathcal{N}(v_t; 0, \sigma_v), \tag{13}$$

where $\mathcal{N}(\cdot; 0, \sigma)$ is a zero-mean Gaussian function with variance $\sigma^2$. We define the *goodness* of the screen as the likelihood ratio

$$S_{screen} \triangleq \frac{\mathcal{L}(screen|d_t, v_t)}{\mathcal{L}(screen|0, 0)}, \tag{14}$$

where $\mathcal{L}(screen|d_t, v_t)$ is the likelihood of the screen given the current distance and velocity values $(d_t, v_t)$ of the interacting players and $\mathcal{L}(screen|0, 0)$ is the likelihood of an ideal screen.

The proximity parameter $\sigma_d$ in (14) is set to the value $\sigma_d = 1$ meter, as in the case of formation detection. The velocity parameter $\sigma_v$ which determines the velocity of player which is "*still enough*" is set to $\sigma_v = 0.5$m/s in our implementation. In our experience an interaction between two players can be interpreted as a screen if the value $S_{screen}$ (14) exceeds the value of 0.8.

***Move.*** A move of a player is transcribed into semantic description simply by outputting the appropriate symbol when player moves from one court area to another. However, to reduce the number of symbols generated when player is moving along the boundary between the two regions (the problem, which can be exacerbated by trajectory noise), an additional hysteresis of 0.5 meters is applied to the output of the transition detector. This way the transition is observed only if player moves at least half a meter into the new region.

### *2.2.4 Template based activity recognition*

By applying the detectors of key elements to the trajectory segment, we get a sequence of symbols - a semantic description of the activity on the court for that particular interval of time. For illustration, Table 1 shows the semantic description of a segment of trajectories from Figure 3, obtained with the *key element detectors* described in the previous section.

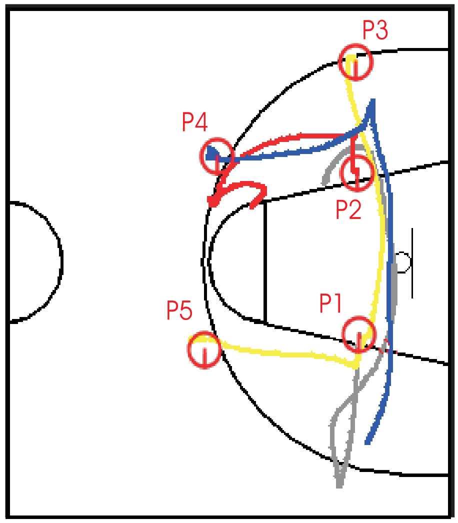



Fig. 3. Example trajectories of a "Flex offense". The numbers denote the players.

Table 1
Initial part of semantic description of trajectories from Figure 3. The numbers in the parentheses explicitly state the order of symbols.

| | | |
|---|---|---|
| (1) M_P1_RIGHT_POST, | (7) FORM_flex1, | (12) M_P4_LEFT_POST_LEFT_WING, |
| (2) M_P2_LEFT_POST, | (8) M_P3_LEFT_POST_LEFT_CORNER, | (13) SCR_P2_P4_LEFT_POST, |
| (3) M_P3_LEFT_CORNER, | (9) SCR_P2_P3_LEFT_POST, | (14) FORM_flex1 inv, |
| (4) M_P4_LEFT_WING, | (10) M_P1_RIGHT_CORNER_RIGHT_POST, | (15) M_P2_LEFT_WING_LEFT_POST, |
| (5) M_P5_RIGHT_WING, | (11) M_P3_KEY_LEFT_POST, | (16) SCR_P2_P3_LEFT_POST ... |

Semantic description of a trajectory segment is used to compare the trajectory segment with the activity template. The matching process is done in two steps.

- In a template, players are denoted by abstract numbers, from 1 to 5. However, these numbers bear no relation to the indexes of players in the semantic description of the observed trajectory segment, due to the abstract nature of activity templates. For this reason, we need to determine the relations between players on the court and the player indexes in the activity template. We call this process *player casting*.
- After the roles of individual players are obtained, we can determine, which of the template-generated semantic descriptions is the most similar to the trajectory-generated description.

***Player casting.*** With player casting, we solve an important combinatorial problem, which some researchers did not address. For example, both Intille and Bobick [25] and Jug [18] performed this task manually. Besides solving the obvious problem of using general activity templates on real world data, player casting increases the robustness of the analysis process. Even if the players are swapped during the tracking process (for example because of the difficult tracking conditions, as described by Kristan [26]),we can still obtain the correct results (with, of course, wrongly assigned identities), as long as identity swaps are not so frequent to completely blur the activity of individual players. The player casting is also needed to address changes in team makeup during the match. Even the actual roles of players on the court may change between the consecutive segments, as players are trained to do this in order to confuse their opponents.

To solve the problem of casting, we have to test 120 (5!) possible casts, and select the permutation, which yields the smallest distance to the semantic description of the selected activity template. Casting has to be done separately for each trajectory segment, and it has to be repeated for each of the activity templates. For each possible permutation (cast), we first generate five distinctive *player agendas*, one for each of the players. One set of agendas is extracted from the trajectory-generated semantic descriptions, and the other set from the template-generated semantic descriptions. This way, the original "combined" activity description is decomposed to five individual activity descriptions - one for each player.

Each player agenda contains only those activities, in which the player in question has participated. For example, starting formation is contained in all five player agendas, while the screen is contained only in two player agendas - in the agendas of those two players who performed the screen. Finally, each individual player move ends up in a single player agenda. Considering these rules, we can transform the semantic description from Table 1 to the five agendas, as shown in Table 2.

Table 2
Five player agendas built from the semantic description in Table 1. The numbering of agendas is consistent with player numbering in previous examples.

| ag1 | ag2 | ag3 |
|---|---|---|
| (1) M_RIGHT_POST | (1) M_LEFT_POST | (1) M_LEFT_CORNER |
| (2) FORM_flex1 | (2) FORM_flex1 | (2) FORM_flex1 |
| (3) M_RIGHT_CORNER_RIGHT_POST | (3) SCR_LEFT_POST | (3) SCR_LEFT_POST |
| (4) FORM_flex1 inv ... | (4) SCR_LEFT_POST | (4) M_KEY_LEFT_POST |
| | (5) FORM_flex1 inv | (5) FORM_flex1 inv |
| | (6) M_LEFT_WING_LEFT_POST | (6) SCR_LEFT_POST ... |
| | (7) SCR_LEFT_POST... | |

| ag4 | ag5 |
|---|---|
| (1) M_LEFT_WING | (1) M_RIGHT_WING |
| (2) FORM_flex1 | (2) FORM_flex1 |
| (3) M_LEFT_POST_LEFT_WING | (3) FORM_flex1 inv... |
| (4) SCR_LEFT_POST | |
| (5) FORM_flex1 inv... | |

In the process of generating the player agendas, we sacrificed the explicit information about the inter-player temporal relations. At this point, only the information about multi-player activity which is implicitly contained in the encoding of screens and formations is retained.

The 120 possible player casts, one for each possible permutation of five players, are then tested by cross-comparing five template-generated player agendas to five trajectory-generated ones. The cross-comparison of players to roles yields similarity matrix $A = [a_{ij}]_{N\times N}$, where element $a_{ij}$ indicates how well the $i-th$ player fits into the $j^{th}$ template role.

In our work, we use the modified Levenstein distance [27] as a similarity measure between the two sequences of symbols. The distance measure assigns a penalty of 2 for each symbol insertion or deletion, and is normalized with the sum of lengths of both symbol sequences. This way, the final output of cost function is normalized to the interval [0..1], yielding zero for a *perfect match* and one for at a *total mismatch*.

The overall cost ($S$) of particular player cast (permutation) is calculated as the sum of individual cost functions ($a_{ij}$) for all five players.This process is repeated for each of the 120 possible permutations – casts ($C$), and the permutation, which yields the smallest cost function ($S(C_{min})$) is accepted as the best explanation of player roles in the observed trajectory segment, when comparing it to the chosen activity template.

$$S(C_{min}) = min\{\sum_{i=1}^{N} a_{ij}\}; \quad j = C_l(i), \quad l = 1 \ldots 120. \tag{15}$$

In our example, the similarity matrix $A$ is

$$A = \begin{bmatrix} 0.22 & 0.81 & 0.66 & 0.77 & 0.59 \\ 0.45 & 0.32 & 0.99 & 0.14 & 0.87 \\ 0.28 & 0.19 & 0.94 & 0.40 & 0.77 \\ 0.91 & 0.95 & 0.31 & 0.49 & 0.89 \\ 0.48 & 0.71 & 0.72 & 0.84 & 0.24 \end{bmatrix},$$

and the correct cast is represented as $C_l = \{1, 4, 2, 3, 5\}$, which produces the smallest distance of $S(C_l) = a_{11} + a_{24} + a_{32} + a_{43} + a_{55} = 1.1$.

***Assigning the label to the analyzed trajectory segment.***After the player casting is completed, the symbols in the trajectory-generated sequence are updated to reflect actual player assignment, and the overall cost function is calculated by comparing the trajectory-generated sequence to the template-generated one, using the previously described modified Levenstein distance.

This process, complete with the role casting, has to be done separately for every activity template from activity database.

After the results of comparison are obtained, the observed trajectory segment is assigned the label of the template that generated the smallest distance to the sequence, generated from the trajectory segment. However, this is done only if the cost is smaller then the certain threshold, which was, in our case, set to 0.6 (see Section 3 for more details). We assume that there is no match for the observed activity in the activity database if the distance is above this value. As an illustration, Figure 4 shows the results of matching the trajectory segment, manually labelled "Flex offense" to 17 different templates, including the "Flex offense" template.

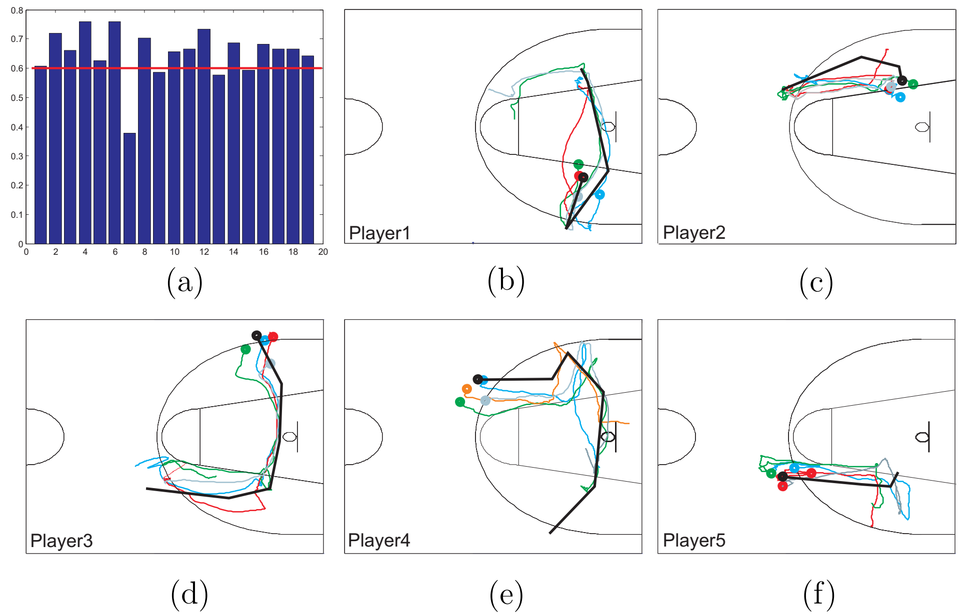



Fig. 4. Matching the trajectory segment, manually labeled as "flex offense" to 17 different action templates, with "Flex offense" template among them. (a) The cost function after the comparison. The correct match can be clearly seen in the form of the lowest bar. Red line represents the threshold of 0.6 (b) to (f) Ideal player paths as defined in the "Flex offense" template (black straight line) and successfully matched player trajectories of four repeatedly performed "Flex offense" activities.

### *2.2.5 Assessment of the team performance*

The described framework for analyzing trajectory data can be used for other tasks than just plain recognition and labeling of team activity. In particular, since we obtain the similarity measure between the trajectory segment and the most similar activity template (in form of modified normalized Levenstein distance between two sequences of symbols), we can interpret it as objective measure of team performance.

The value of this measure is influenced by the execution of the required key elements, the order of execution, their position on the court, and the coordination between team members during the execution of the particular play. Low values of this similarity measure indicate that all of the elements were executed at the right places and that the players were well coordinated. High values may indicate poor team performance, which may be attributed to multiple causes. In case of training, it could suggest that the team did not learn or train the particular activity well enough. In the case of actual match, it could indicate that the team is facing successful defense of the opposing team, which manages to destroy their offensive play (and therefore the value can be interpreted as the measure of quality of the opposing teams' defensive tactics). Finally, low values may suggest that the team is not playing in a organized manner at all, either because the opponents are able to completely block their attempts to organize activity, or because the opponents are weak enough that the team can score even without any particular play organization.

## 3 Experiments and results

Our approach is general enough to be applied to any kind of (sufficiently accurate) trajectory data. However, to test the approach, we obtained the trajectory data using computer vision based tracking methods. We detail our experimental setup here to document the nature of the input data, on which our trajectory based activity analysis methodology was tested.

### *3.1 Experimental setup*

The structure of our experimental system is shown in Figure 5. In order to perform trajectory analysis, we recorded a number of test videos, which included both a real match and training-level play under a supervision of a basketball coach. To obtain player trajectories, tracking has been performed on those videos.

We generated the needed activity templates using our own proprietary software (”play designer”), which provides a graphical user interface, with the functionality similar to the board a basketball coach uses to explain their ideas to the players. The software is able to automatically render activity templates both to their graphical representation and to their semantic descriptions.

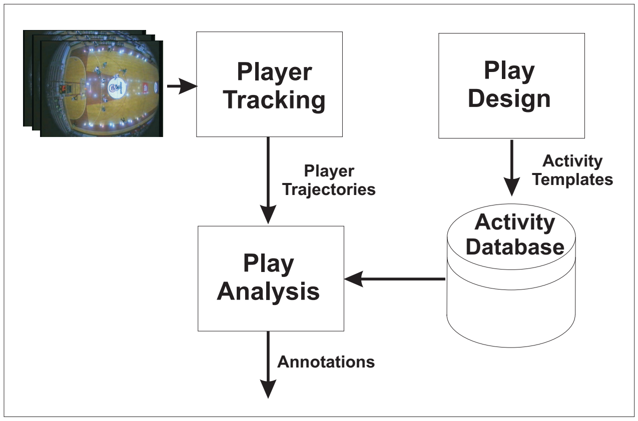



Fig. 5. System overview.

### *3.1.1 Video acquisition and tracking*

To obtain the videos of actual basketball play, we used two cameras, permanently mounted to the ceiling of the sports hall, in which basketball matches are played. Such camera setup enabled us to obtain videos with birds' eye view of the play, and two cameras are sufficient to cover the whole court for the whole duration of the match. Image from one of the cameras is shown in Figure 6.

Two sets of videos have been recorded:

- Set A: A basketball match on the national championship level, which was used for experiments regarding the trajectory segmentation.
- Set B: A set of videos, where a team of players was repeatedly executing three well known types of basketball offense under the supervision of the coach, and other team was executing a passive defense (defending the basket, without trying to start their own offense). Both videos with and without defensive team have been recorded.

To obtain motion data, we performed operator-supervised tracking on these sequences. We used modified color histogram based *CONDENSATION* algorithm [26, 28] as the tracking engine, built into the user friendly graphical interface. An operator was supervising the tracking and corrected any errors that appeared during the tracking process. Tracking was coupled with the appropriate calibration, which provided the mapping of image coordinates to the real-world (court) coordinates and compensated for radial distortion, originally present in video data. The tracking yielded a measurement of positions for all of the players in each of the video frames. Since we used PAL video cameras, our tracking system provided trajectory data for each player with 25 samples per second. At the end of the tracking, data were smoothed using a 25 samples wide symmetric Gaussian kernel to reduce the jitter in trajectories.

Figure 2 shows operator's view of the tracking process. Based on our setup and our previous research [29, 30] we estimate that the obtained player positions contain RMS error in the range between 0.3 and 0.5 meters. Due to the nature of the play (majority of the play takes place directly under the camera) the error is closer to 0.3 than to 0.5 meters.

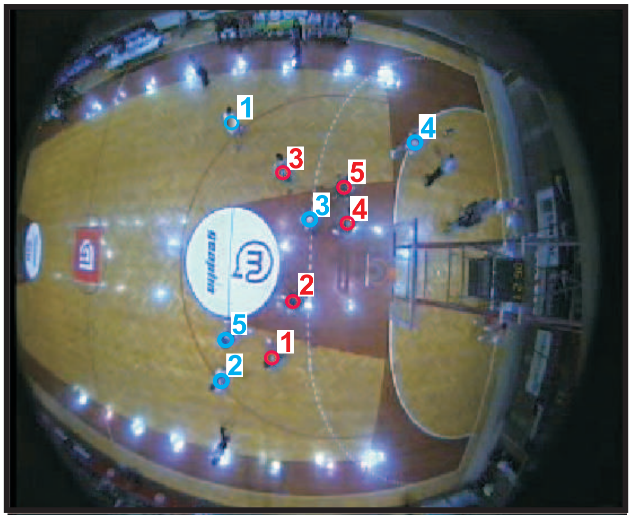
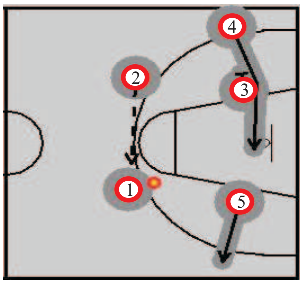


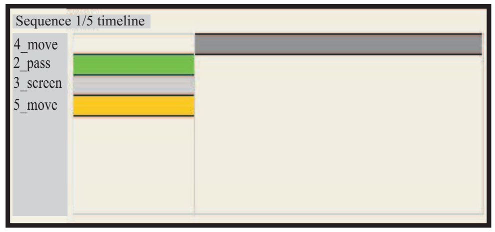

Fig. 6. Operator-supervised tracking in progress.

### *3.1.2 Generating activity templates*

We represent expert knowledge in the form of activity templates, which have been modeled after the conventional tools, which coaches use to present plays to the players. The developed graphical interface (*play designer*) was used to graphically place the *key events* on the basketball court, associate them with individual players, and define the order and timeline of key element execution. Complex activities (such as more complex types of basketball offense) are represented by multiple sequences in one template, which follow one after another. The tool is able to automatically generate the required semantic descriptions, used in trajectory analysis. Semantic descriptions represent only a subset of information from each activity template, as only three key elements of the basketball play are used in the analysis process. During the rendering of semantic descriptions, absolute placement of key elements is replaced by court regions (given the court partitioning, as described in Section 2.2.2) and the intervals between the elements are discarded, retaining only their order.

Figure 7 shows the spatio-temporal representation of the first sequence from the activity template called "Flex offense" [31]. On the left side, starting positions of the players and their ideal paths are shown. On the right, the timeline and relative durations of every event are presented. It can be observed that player 5 should move behind the three point line at the same time as player




(a) (b)

Fig. 7. Spatial representation (a) and timeline (b) for the first sequence from the ”Flex offense” template.

3 prepares the screen for player 4 and player 2 passes the ball to player 1. Finally, after players 2, 3 and 5 performed their actions, player 4 should move close to player 3.

### *3.2 Experiments on trajectory data*

Several experiments have been performed to test different aspects of the proposed methodology. We used two sets of trajectory data from two sets of videos, A and B, as described before. Set A was used to evaluate the trajectory segmentation method, and set B was used to evaluate the recognition of three types of basketball offense.

#### *3.2.1 Temporal trajectory segmentation*

In the first set of experiments, we tested the performance of our trajectory segmentation method. Trajectory set A, which was obtained from the videos of the regular match, was used for this purpose. 40 minutes of video at 25 frames per second yielded 60700 trajectory samples for each of the 10 players (five from each of the teams). To obtain gold standard, we manually labeled all transitions between offensive, defensive and time-out phases, as seen from the perspective of one of the teams. For other team, we inverted the labels for offense and defense, while the time-out labels remained the same. These annotations were based on our visual observations of the videos from set A. We used the changes in ball possession to determine the exact placement of each annotation with high degree of objectivity.

This way, each trajectory sample has been assigned one of three labels (offense, defense, timeout). The trajectories and the labels were fed to the algorithm, described in Section 2.1. The success rate of the described automatic segmentation method has been measured as the percentage of time (which equals the percentage of samples) in which the resulting label was consistent with the manual annotations.

To test the generalization capability of the method, we performed four tests. In the first two tests, the trajectories from one team were used for training, and the trajectories of the other team were used for testing. In the second two, training and testing was performed on the same set of data, to get an estimate of the upper limit of the classification rate. Table 3 shows the results.

Table 3
Results for the temporal trajectory segmentation.

| | Training sequence | |
|---|---|---|
| Test sequence | Team 1 | Team 2 |
| Team 1 | 93.53 % | 90.83 % |
| Team 2 | 91.35 % | 90.92 % |

The model exhibited good generalization properties, regardless of the training and testing set, as the recognition rates did not drop significantly when the data from the opposite teams were used for training and testing. However, it can be observed that slightly better results were obtained when the trajectories of Team 1 were used for training, regardless of the test set. One of the likely explanations for such result is that the manual annotations were based on the activity of the first team. Following the basketball theory, the annotations for the second team should be the exact inverse, since at each moment, one team is in offense, and the other is in defense. However, in practice, there is short period between the loss of the ball and the organized change in tactics of both teams. Therefore, attaching the common labels to the ball possession of one team produces a slight asymmetry in labels. Further analysis also revealed that the model often failed in the case of time-outs, which is not surprising, as there were relatively few training examples of this phase (only four one-minute-long timeouts). Figure 8 shows the segmentation results on trajectory data.

### *3.3 Activity recognition*

Tests of the segmentation methodology on trajectories from the set A confirmed that the described approach could be used for automatic trajectory segmentation. Therefore, in the rest of the experiments, we limited ourselves to the second set of data (set B), which included manually segmented trajectory sequences, obtained from the videos, which contained multiple repetitions of three well known types of basketball offense. The data from set B contained 71 trajectory segments, of which 39 were played with and 32 without the defensive team. Trajectory segments contained only the player trajectories from one team - from the team that played the offensive role. This way, we ensured

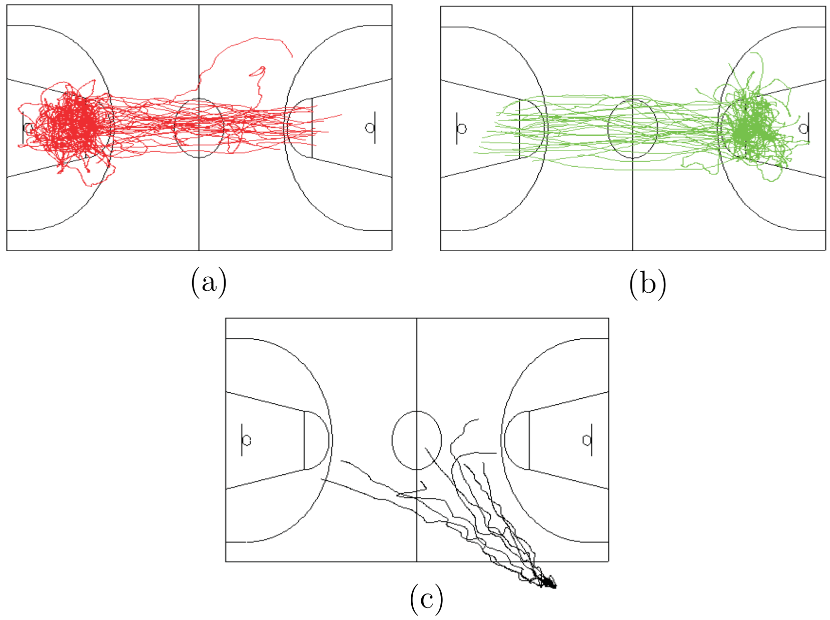


Fig. 8. Segmentation results: (a) Offensive phase, (b) Defensive phase, (c) Time outs.

the consistency and density of information in the test data. If, on the other hand, the regular league match data would be used, the systematic testing would be impossible due to small number of repetitions and wide variety of activities performed during the match.

Trajectory segments were matched to 34 different action templates (left and right variants of 17 different offensive activities), which were obtained from various sources, such as basketball literature [2], and consultation with the local coach. Of the 34 templates available, 6 corresponded to the three types of activities contained in the test data.

Using this set of trajectory segments resulted in 100% correct recognition - all of the trajectory segments yielded shortest modified Levenstein distance for the correct activity template. To further test the robustness of our approach, we performed two additional experiments.

First, we tested the influence of spurious symbols on the recognition rate. These symbols are almost always present in the semantic descriptions, due to spurious detection of key elements and poorly executed player activities. The better the opposing team, more spurious symbols will be present in the trajectory-generated description of the observed team, as the opposing team will be able to disrupt the execution of observed team activity. Resistance to spurious symbols increases the usefulness of the described method in the regular match analysis, where the organized activity is usually intermixed with significant quantities of random activity.

To perform this test, we added different ratios of additional symbols to the original symbol sequences. Added symbols were randomly selected from the symbol dictionary, and were inserted into the original description at random positions. To compensate for effects of random draw, we repeated every experiment 20 times, making in total 1420 runs ($20 \times 71$ trajectory segments) of the activity recognition algorithm for each chosen amount of additional symbols. Table 4 shows the recognition results after the different amounts symbols were added.

Table 4
Average recognition success rate when different amounts of spurious symbols were added. Every experiment was repeated 1420 times for each percentage of the added symbols.

| Percentage of added symbols [%] | 0 | 50 | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|---|---|
| Recognition rate [%] | 100 | 99.0 | 98.9 | 97.5 | 96.2 | 94.8 | 91.8 |

It can be observed that the proposed recognition procedure is extremely robust to the added random symbols. This is due to the nature of similarity measure used (the modified Levenstein distance) which is calculated both from the presence of symbols and their order. By adding reasonable amounts of symbols, the distances to all templates are increasing roughly in the same manner. Given the nature of basketball play, there is little chance that such random motion would correspond to any template definition of an organized play, unless extremely short templates are used.

In the second experiment, we tested the separability of test data, using our approach for activity recognition. This was done by cross comparing the trajectory generated semantic descriptions of each of the 71 trajectory segments, and calculating the average distance for the matched and mismatched pairs of segments separately. This way, we obtained a confusion matrix for the three types of basketball offense. The matrix is shown in Table 5.

Table 5
The confusion matrix - the average distance values for the three types of basketball offense.

| | Class of action | | |
|---|---|---|---|
| Class of action | 52 | Flex | Moving stack |
| 52 | **0.3205** | 0.6340 | 0.7198 |
| Flex | 0.6340 | **0.4893** | 0.6998 |
| Moving stack | 0.7198 | 0.6998 | **0.4667** |

In the third experiment, we used all available templates to perform activity recognition, and calculated average distance when comparing each segment to the matching template and to 33 mismatched templates. Results are shown in Table 6).

Table 6
The confusion matrix - the cost values of comparison of test examples to templates.

| | Class of template | |
|---|---|---|
| Class of action | Matching template | Average of 33 mismatched templates |
| 52 | **0.3695** | 0.6442 |
| Flex | **0.3847** | 0.6757 |
| Moving stack | **0.5106** | 0.7111 |

The last two experiments suggest that there is a significant difference between the distances to matching templates (or matching trajectory segments) and the distances to mismatched templates (or mismatched trajectory segments). This shows that our methodology for activity recognition generates feature values, which are well separable. In our case, the recognition threshold could be set to any value between 0.51 and 0.64 to obtain best recognition rate.

## 4 Conclusion and future work

We presented a two step approach to the analysis of basketball game. Our ultimate goal is automatic segmentation of trajectory data into meaningful play segments (offense, defense and time outs) and automatic recognition of team activity from those segments.

We have demonstrated that by observing only the average position of all players in the team, it is possible to segment the basketball game with reasonable accuracy. We modeled every phase as a two-component Gaussian mixture model, with one component representing the transition to a particular phase and the other representing the main behavior during that phase. To determine the model parameters we applied the EM algorithm to the set of manually labeled trajectory data.

It was shown that the obtained model is general enough to be trained on one team and used on another. This suggests that it could be possible to derive and train a general basketball play model, which could be used to successfully segment trajectories into the game phases without any tuning.

In the second part of the paper, we presented a method for the automatic recognition of specific basketball activities. The method detects basic elements of the basketball play, and based on the detected elements, it transforms the trajectories into semantic descriptions of the observed team activity. The resulting symbol sequences are essentially a narrative of the activity on the court. Such approach turned out to be extremely robust with regard to the spurious symbols in the descriptions. We demonstrated the consistent behavior of the described approach on our test database of 71 repetitions of three types of basketball offense.

The presented approach assumes that we already have descriptions of the activities of interest in the form of activity templates. However, there are cases, where such approach is not entirely appropriate. If the objective is to study the activity of the opposing team, or to obtain coarse statistics on team behavior, it is unreasonable to assume the templates that cover all the interesting behavior would be available. Nevertheless, the presented framework could be extended to cover this case as well. Our method does not depend on how the reference descriptions are obtained. In our case a graphical tool to design activity templates was used; however, in the absence of better options, the reference descriptions could be extracted from the trajectories as well. This opens a possibility that an expert (coach) chooses a particular action of interest, whose repetitions are then found automatically by the software among all the segments from the whole match. Finally, performing such search with each trajectory segment as template (undoubtedly a computationally intensive task) could give us the clustering of activities, which would essentially represent the description of common team behavior.

The future work will focus on three problems. First, in many cases during a real match, the activity inside a particular phase is restarted (e.g. after a foul or a rebound), and no phase change is present. Our temporal segmentation framework does not cover the segmentation in such cases. To deal with such scenarios, the activity recognition framework should be extended to search for multiple activities and their temporal boundaries inside one phase.

Second, the activity recognition framework should be extended to take into the account the importance of individual roles that players have in the observed activity. In basketball theory, some elements of the complex activity are usually more important than others, and some player roles are more important than others. This is reflected in team performance, as players will put more effort in the successful execution of critical tasks, and less effort in the exact execution of other elements.

Finally, we anticipate that the inclusion of the information about the ball possession should considerably improve the performance of segmentation and recognition phase. However, with the current state of technology, such solution would require a significant amount of manual work to obtain such data, and does not fit well into the concept of automated analysis framework designed to save time.

## References


[1] I.M. Franks and G. Miller. Eyewithness testimony in sport. *Journal of sports behavior*, 9:39–45, 1986.

[2] J. Kresse and R. Jablonski. *The Complete Book of Man-To-Man Offense.* Coaches Choice, 2nd edition, 2004.

[3] D.M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU, Academic Press*, 73(1):82–98, 1999.

[4] C. Cedras and M.A. Shah. Motion-based recognition: A survey. *Image and Vision Computing (IVC)*, 13(2):129–155, March 1995.

[5] P. Xu, L. Xie, S. Chang, A. Divakaran, A. Vetro, and H. Sun. Algorithms and systems for segmentation and structure analysis in soccer video. In *IEEE International Conference on Multimedia and Expo (ICME), Tokyo, Japan*, pages 184–187, 2001.

[6] H. Pan, P. Van Beek, and M.I. Sezan. Detection of slowmotion replay segments in sports video for highlights generation. In *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing*, 2001.

[7] R. Urtasun, D.J. Fleet, and P. Fua. Monocular 3d tracking of the golf swing. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 932 – 938, June 2005.

[8] A.A. Efros, A.C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Ninth IEEE International Conference on Computer Vision (ICCV), Nice, France*, volume 2, pages 726–733, 2003.

[9] Y. Luo, W. Tzong-Der, , and H. Jenq-Neng. Object-based analysis and interpretation of human motion in sports video sequences by dynamic Bayesian networks. *Computer Vision and Image Understanding*, 92(2-3):196–216, 2003.

[10] D. Gavrila and L. Davis. 3d model-based tracking of humans in action: A multi-view approach. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, San Francisco*, pages 73–80, 1996.

[11] R. Cassel. Acrobatic movement analysis based on computer vison. In *CVBASE'06, Procceding of ECCV workshop on Computer Vision Based Analysis in Sport Environments*, pages 83–91, May 2006.

[12] M. Dimitrijevic, V. Lepetit, and P. Fua. Human body pose recognition using spatio-temporal templates. In *ICCV workshop on Modeling People and Human Interaction, Beijing, China*, volume 2, October 2005.

[13] L.W. Campbell and A.F. Bobick. Recognition of human body motion using phase space constraints. In *ICCV'95*, pages 624–630, 1995.

[14] A. Wiloson and A.F. Bobick. Learning visual behavior for gesture analysis. In *Proceeding of the IEE Symposium on Computer Vision, Florida, USA*, pages 19–21, 1995.

[15] H. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, 14:609–615, 1996.

[16] H. Johnson and D. Hogg. Represention and synthesis of behavior using

gaussian mixtures. *Image and Vision Computing*, 20:889–894, 2002.

[17] S.S. Intille and A.F. Bobick. Recognizing planned, multiperson action. *Computer Vision and Image Understanding: CVIU*, 81(3):414–445, 2001.

[18] M. Jug, J. Perš, B. Dežman, and S. Kovačič. Trajectory based assessment of coordinated human activity. In *Computer Vision Systems, Proceedings of Third International Conference ICVS 2003*, pages 534–543, Graz, Austria, April 2003.

[19] F. Li and R.J. Woodham. Analysis of player actions in selected hockey game situations. In *Proceedings of the The 2nd Canadian Conference on Computer and Robot Vision (CRV'05), Canada*, pages 152–159, 2005.

[20] H.L. Harkins and J. Krause. *Motion Game Offenses - For Men's and Women's Basketball*. Coaches Choice, 3 edition, 2001.

[21] W.S. Erdmann. Gathering of kinematic data of sport event by televising the whole pitch and track. In *Proceedings of 10th ISBS symposium, International Society of Biomechanics in Sports, Rome, Italy*, pages 159–162, 1992.

[22] R.A. Redner and H.F. Walker. Mixture densities, maximum likelihood and EM algorithm. *SIAM Review 26*, 195-202, 1984.

[23] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., New York, 1997.

[24] R. O. Duda, P. E. Hart, and D.G. Strok. *Pattern Classification*. John Wiley & Sons, Inc., 2 edition, 2000.

[25] S.S. Intille and A.F. Bobick. A framework for recognizing multi-agent action from visual evidence. In *in: Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence, Orlando, Florida, United States*, pages 518–525, July 1999.

[26] M. Kristan, J. Perš, M. Perše, M. Bon, and S. Kovačič. Multiple interacting targets tracking with application to team sports. In *4th International Symposium on Image and Signal Processing and Analysis ISPA*, pages 322–327, September 2005.

[27] V. Makinen, G. Navarro, and E. Ukkonen. Transposition invariant string matching. In *Proc. 20th International Symposium on Theoretical Aspects of Computer Science (STACS 2003), in: Lecture Notes in Comput. Sci., vol. 2607, Springer-Verlag.*, page 191202, 2003.

[28] M. Perše, J. Perš, M. Kristan, G. Vučkovič, and S. Kovačič. Physics-based modelling of human motion using kalman filter and collision avoidance algorithm. In *International Symposium on Image and Signal Processing and Analysis, ISPA05, Zagreb, Croatia*, pages 328–333, September 2005.

[29] J. Perš and S. Kovačič. Tracking people in sport: Making use of partially controlled environment. In *Computer analysis of images and patterns: 9th international conference, CAIP 2001 , Lecture notes in computer science*, volume 2124, pages 374–382, September 2001.

[30] J. Perš, M. Bon, S. Kovačič, M. Šibila, and B. Dežman. Observation and analysis of large-scale human motion. *Human Movement Science*, 21(2):295–331, July 202.

[31] R. Righter. *Flex: The Total Offense*. Championship Books, Ames, Iowa, USA, 1984.