Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video

Conference Paper · November 2003

DOI: 10.1145/957013.957018 · Source: DBLP

6 authors, including:

Hon Wai Leong
National University of Singapore

Kongwah Wan
Institute for Infocomm Research

Trajectory-Based Ball Detection and Tracking with Applications to Semantic Analysis of Broadcast Soccer Video

[1]Xinguo Yu, [1]Changsheng Xu, [2]Hon Wai Leong, [1]Qi Tian, [1]Qing Tang, and [1]Kong Wah Wan

[1]Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
{xinguo, xucs, tian, tracey, kongwah} @ i2r.a-star.edu.sg

[2]School of Computing, National University of Singapore, Singapore 117543
leonghw@comp.nus.edu.sg

ABSTRACT

This paper first presents an improved trajectory-based algorithm for automatically detecting and tracking the ball in broadcast soccer video. Unlike the object-based algorithms, our algorithm does not evaluate whether a sole object is a ball. Instead, it evaluates whether a candidate trajectory, which is generated from the candidate feature image by a candidate verification procedure based on Kalman filter, is a ball trajectory. Secondly, a new approach for automatically analyzing broadcast soccer video is proposed, which is based on the ball trajectory. The algorithms in this approach not only improve play-break analysis and high-level semantic event detection, but also detect the basic actions and analyze team ball possession, which may not be analyzed based only on the low-level feature. Moreover, experimental results show that our ball detection and tracking algorithm can achieve above 96% accuracy for the video segments with the soccer field. Compared with the existing methods, a higher accuracy is achieved on goal detection and play-break segmentation. To the best of our knowledge, we present the first solution in detecting the basic actions such as touching and passing, and analyzing the team ball possession in broadcast soccer video.

Categories and Subject Descriptors

I.4.8 [**Scene Analysis**]: *Color, Shape, Tracking.*

I.2.10 [**Vision and Scene Understanding**]: *Video analysis.*

General Terms

Algorithms, Performance, Experimentation.

Keywords: Ball Detection and Tracking, Trajectory-Based, Semantic Analysis, Event Detection.

1. INTRODUCTION

Soccer video analysis is receiving increasing attention from researchers. This interest is motivated by its possible applications over a wide spectrum of topics [1, 4-6, 13-15], e.g. indexing, summarization, video enhancement, tactics analysis, performance analysis, object-based encoding, and content-based streaming. The primary benefits include quick and accurate access to video segments of interest and enhancement of the overall video quality. We need to automatically and efficiently analyze soccer video because its size is huge and the value of sports video (including soccer video) drops significantly after a relatively short period [1].

More keenly than ever, researchers desire to obtain the position of the ball for each frame for broadcast soccer video [3, 6, 11, 16-17] because the information on ball positions over frames can greatly enhances soccer video analysis. For example, it can play a crucial role in analyzing video structure, ball possession, event detection, etc. However, the problem of finding the ball position in broadcast soccer video is challenging because of the following [3, 6, 16-17]:

- The appearance of the ball varies irregularly over frames. Its size, shape, color, and velocity change irregularly over frames.
- Many other moving objects are similar in appearance to the ball. For example, many regions of player look like the ball.
- The ball is small.
- The ball is often merged with lines.
- The ball is often occluded by people.

Typical balls in broadcast soccer video (which are obtained by removing the other objects in the selected frames) are shown in Figure 1. These typical balls testify the above-listed challenges.

Figure 1. Typical balls in broadcast soccer video. The balls in (a), (b), and (c) are from closed-up frames; the balls in (d) and (e) are from full view frames.

These challenges lead to a fundamental difference between ball detection-and-tracking in broadcast soccer video and other detection-and-tracking problems in the literature. The key difference is that there is virtually *no property* available to distinguish the ball from other objects *within the same frame*.

A trajectory-based algorithm for detecting and tracking the ball in broadcast soccer video was first proposed in [16-17] by the authors. The key idea in this algorithm is to evaluate whether a ball candidate trajectory is a ball trajectory. It has four components: Ball Size Range Extraction, Non-Ball Object Removal, Ball Trajectory Mining, and Ball Trajectory Extension. In the first component, the algorithm infers the ball size by the people heights in the soccer field. In the second component, it removes the identified non-ball objects in each frame by an object evaluation function. The remaining objects are considered to be the ball candidates. In the third component, it creates the candidate feature images (CFIs) (see [16] for their definitions) and uses a heuristic procedure to reduce false candidates in CFIs. A Kalman filter-based candidate verification procedure generates candidate trajectories from CFIs. A value is computed for each candidate trajectory to indicate how likely it is a ball trajectory. Then, a procedure is used to choose the ball trajectories according to their values. In the fourth component, each ball trajectory is extended by a Kalman filter-based template matching procedure and an interpolation procedure. This algorithm achieved about 85% accuracy for finding the ball.

The contributions of this paper are as follows: Firstly, we improve the algorithm presented in paper [16-17]. The improved algorithm, whose block diagram is depicted in Figure 2, has new four components: Ball Size Estimation, Candidate Detection, Candidate Trajectory Generation, and Trajectory Processing. More importantly, it has improved techniques used in each component. Three main improvements are the following. (1) To identify the ball trajectory better, it uses an iterative procedure to obtain ball trajectories. (2) To improve the ball size estimation, besides finding the people sizes we also find the sizes of the goalmouth and the ellipse (which is the projection of the center circle in the soccer field). (3) To properly filter non-ball objects by size, the improved algorithm estimates the ball size for each point in the frame instead of one size for whole frame. These improvements make the algorithm more robust and enhance the accuracy of finding the ball from about 85% to above 96%. Our approach for finding the ball position is rooted in ideas from probability and statistics, motion theory, and computer vision. Secondly, we propose a new approach for analyzing the soccer video which is based on the ball trajectory computed rather than the low-level feature. This approach can improve play-break analysis, high-level semantic event detection. More importantly, it can detect some basic actions such as touching and passing and analyze team ball possession, which cannot be analyzed based only on the low-level feature.

2. RELATED WORK

2.1. Related Work on Object Tracking

There have been many detection and tracking algorithms proposed over the last two decades. These algorithms can be classified into three categories: feature-based, model-based, and motion-based.

In *feature-based algorithms*, some *features* are employed to discriminate targets from other objects within a frame to characterize targets in a property state space. For example, parameterized shapes [3], color distributions [4], shape and color together [10] are often used in target representations. *Model-based algorithms*, however, use not only features but also *high-level semantic representations* and *domain knowledge* to discriminate targets from other objects [9]. *Motion-based algorithms*, on the other hand, rely on the *methods for extracting and interpreting motion consistencies over frames* to segment the moving object [8].

In all three cases, the target identification is performed *within a frame* using measurements provided by properties of the target, although motion properties may relate to several frames. These methods could be called object-based since their crucial step is to evaluate whether a detected object is a target. They all have three main elements: target representation, property extraction, and discrimination. These object-based methods have an implicit assumption that targets are somehow different from other objects within a frame. A detection and tracking problem is called an object-distinguishable problem if its targets are different from other objects in some properties except their positions within a frame. Otherwise it is called an object-undistinguishable problem.

2.2. Related Work on Soccer Video Analysis

Soccer video analysis is an active topic. In this topic, prior work has been on structure analysis [13-14], event detection [5, 11], presentation [15], shot classification [5], summarization [5], and ball detection and tracking [3, 6, 15-17]. Most existing papers analyze the video based on low-level features as they cannot obtain the accurate ball trajectory [3-6, 15]. One major analysis model for the sports video, including soccer video, has three main steps [4-5]. The first step cuts the video into shots. The second step classifies the shots. The last step detects the game events by shot change templates. Most of the results are not very accurate as these templates do not have the exact correlation with game events. So far little work analyzes the soccer video based on the ball trajectory [11]. Tovinkere and Qian proposed a hierarchical entity-relationship model capturing the domain knowledge of soccer [11]. They defined basic actions and complex events, and used a set of nested rules to detect the occurrence of a certain event. Since their system needed 3D position of the players and the ball as input, it cannot be applied to analyze broadcast soccer video because it does not contain 3D information. Although the authors claimed that their system could detect exhaustive set of events, only limited amount of evidence was provided, because only one basic action (*deflection*) and one complex event (*save*) were discussed. To conclude, analyzing broadcast soccer video based on the ball trajectory is a new approach. More and more researchers wanted to use the ball trajectory to boost their analysis results [3, 5, 15]. However, in these papers the ball trajectory played a minor role as they could find the ball only for some selected cases. D'Orazio et al. [3] used a modified Circle Hough Transform to detect the ball for selected frames from real video rather than broadcast video. This method can obtain satisfactory results only when the physical ball is in single color and the complete ball occurs in the frame. However, the frequent occurrence of incomplete balls is common in broadcast soccer video, as Figure 1 shows.

2.3. Proposed Method

To solve the object-undistinguishable problem, we need more motion information to discriminate the target from other objects. The work in [8] used motion data to segment moving object, but it could only mine limited motion information from rich motion information of video. Furthermore, it had neither trajectory analysis nor temporal filter. Indubitably, it was an object-undistinguishable problem to detect and track the ball in broadcast soccer video due to its challenges. In a number of frames in broadcast soccer video, some objects such as some regions of people may look like the ball. To complicate the matters, sometimes the ball itself may not be ball-like due to deformation, half blocking, etc. Thus, object-based methods could not find the ball accurately.

To overcome the challenges in finding the ball, we propose a *trajectory-based* algorithm with four components as depicted in Figure 2, which is improved from our previous algorithm presented in [16-17]. In Ball Size Estimation component, the salient objects (goalmouth, ellipse, and people) are detected and tracked. Then we estimate the ball size from the salient objects. With ball size estimation, we overcome the challenges that the ball is small and the ball size varies over frames. In Candidate Detection component, each frame produces its candidates by removing the identified non-ball objects. Thus, the ball and some objects that look like the ball are remained. In Candidate Trajectory Generation component, CFIs are created from all candidates in a sequence of frames. Each CFI produces a candidate trajectory set. In Trajectory Processing component, a trajectory discrimination procedure identifies ball trajectories and removes the trajectories that overlap with the identified ball trajectories. Then, a model matching procedure extends the obtained ball trajectories and the remaining candidate trajectories. The discrimination and extension procedures work iteratively. Unlike the object-based approach, we do not evaluate whether a sole object is a ball. Instead, we evaluate whether a candidate trajectory is a ball trajectory. In a given video segment, each object has its own trajectory. A non-ball trajectory might contain some objects that look like the ball but such objects have a small ratio in the trajectory. On the other hand, a ball trajectory may also contain some objects that do not look like the ball, but we expect that most would be ball-like. Hence, the ball trajectory can be identified reliably.

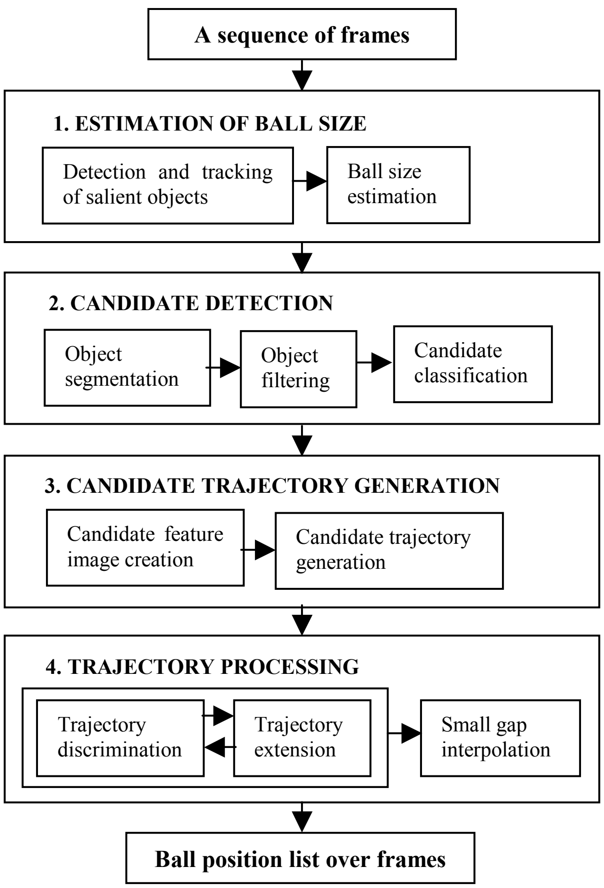


Figure 2. Block diagram of the improved trajectory-based algorithm for detecting and tracking the ball position in broadcast soccer video.

With the ball trajectory computed, we propose a new approach, which is based on the ball trajectory rather than low-level features, to analyze broadcast soccer video. A soccer game essentially consists of a series of player actions and player-ball or player-player interactions. Since a key objective of the players is to control the ball goal-wards, we expect the player-ball interactions to be the most important elements of the game. Some of the player-ball interactions indicate certain meanings, e.g., a start, a stop, a possession obtainment. Some of the actions and interactions lead to certain consequences, e.g., a goal or yellow-card, as determined by the soccer rules. In general, people want to find the semantic events such as goal, corner kick, and free kick. All semantic events must be the consequence of simple physical actions. Hence, we first detect all simple physical actions. Then, we further evaluate whether a simple physical action leads to a semantic event. We begin with detecting the motion pivots of the ball because each simple physical action of player-ball interactions should cause a motion pivot of the ball. For each pivot, we evaluate whether it is a touching point by evaluating whether a person touches the ball. To determine which team touches the ball, we find the team of the person who touches the ball. Other detections such as the goalmouth detection and shooting detection are fused into the various event detection procedures.

3. BALL DETECTION AND TRACKING

Now we describe the four components of the algorithm in turn. However, the procedures that have already been presented in [16-17] are abridged or omitted.

3.1. Ball Size Estimation

We begin with the principles used in ball size estimation. Let R_1 and R_2 be the diameters of the physical ball and any object in the field respectively. Let *b(center)* and *o(center) be* the diameters of the ball image and the object image when they both are at the center of the frame. According to the image generation principle of the pin-hole camera, we have

$$b(center) : o(center) = R_1 : R_2. \quad (1)$$

The ball and the object may not be both at the center of the frame so we need to use the projection matrix to compute the ball size from the object size. We create a simplified projection matrix for each frame type by statistics. In the algorithm, we detect the salient objects and find the frame type according to the detected salient objects. Consequently, for a given frame we can compute the ball size at the center of the frame if we can find some salient objects in it. In the previous algorithm presented in [16-17], we compute the ball size only from the sizes of the people in the field. As a result, the estimated ball size is not very accurate since the sizes of the people are various and the people can not be segmented very accurately. To improve the performance of the ball size estimation, we estimate the ball size from the ellipse and the goalmouth if we can find them. The ball size computed from the ellipse or the goalmouth is accurate, but they together appear in 20-30% frames. We still compute the ball size from the sizes of the people if we can find neither the ellipse nor the goalmouth. Finding these salient objects is easier than finding the ball object directly since they have constant low-level features and useful domain knowledge.

Detection and tracking of the ellipse: In soccer video, the ellipse has three characters. First, a center line (which is almost vertical and very long) is in the middle of the ellipse. Second, the ellipse is almost horizontal. Last, the ellipse is large. We can accurately detect and track the ellipse by using the listed three characters. Figure 3 illustrates the procedure to detect the ellipse in a frame. When the center line is not vertical, we can transform the frame to make the center line to be vertical. As a result, the ellipse becomes horizontal in the transformed image since it is perpendicular to the center line.

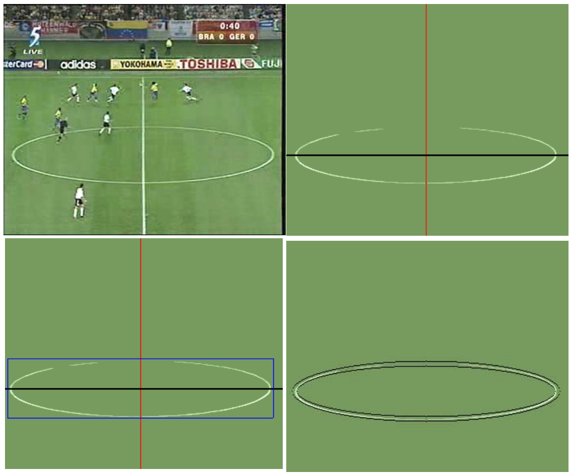


Figure 3. An image with the ellipse, the found center, the found bounding box, and the drawn ellipse ring.

Detection and tracking of the goalmouth: We find the goalmouth if we cannot find the ellipse. The goalmouth has three characters too. First, its two poles are almost vertical. Second, the goal poles and goal bar are bold line segments. Third, the goal poles are short line segments. With the listed three characters, we can find the goalmouth accurately. Figure 4 illustrates the procedure to detect the goalmouth in a frame.

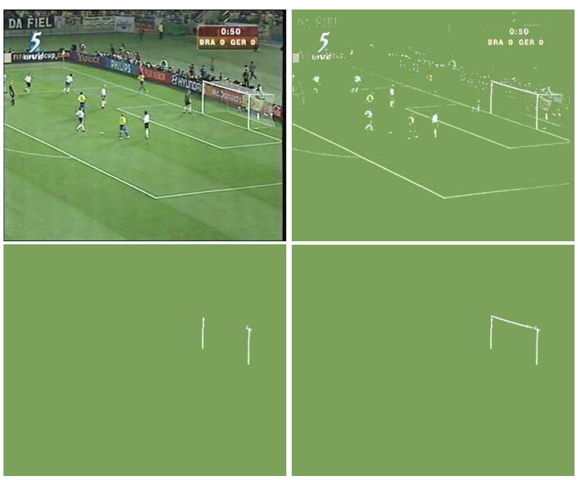


Figure 4. An image with the goalmouth, the image after color segment, the found poles, and the found goalmouth.

Adjusting of the ball size: Once we estimate the ball size for the given video segment, these ball sizes form a function over frame. Among these ball sizes, those estimated by ellipse or goalmouth are reliable. Adjust the ball size through smoothing the curve of the function of the ball size without changing the reliable sizes.

Ball size sieve: We can know the ball size at *(i, j)* in the frame since we have the projection matrix and the ball size at the frame center. Considering the ball deformation and the processing errors, we use the ball size range $s(i,j) = [b(i,j) \bullet (1-\Delta_1), b(i,j) \bullet (1+\Delta_2)]$ replace the ball size $b(i,j)$, where Δ_1 and Δ_2 are the extensions to tolerate the errors. $S = (s(i,j))_{w \times h}$ forms a complete size sieve, where w and h are the width and height of the frame respectively.

3.2. Ball Candidate Detection

In a frame, some non-ball objects may look like the ball more than the ball itself. Hence, instead of building the ball representation, we use the anti-model approach to obtain the ball candidates. We collect all detected objects from a frame and filter this collection by the sieves (anti-models). The remaining objects are considered to be the ball candidates of the frame. Then, we classify the obtained candidates into three categories.

3.2.1. Ball Candidate Generation

Let $\mathrm{O}(F)$ be the set of all object in *F*. We apply the following sieves to the set to obtain the ball candidates of *F*.

Ball Size Sieve Θ_1 : It is built in Section 3.1.

Line Sieve Θ_2 **:** We can filter all long lines (straight lines and curves) since the ball cannot be deformed into a long line.

Ball Color Sieve Θ_3 : The ball must have some pixels with colors falling into the ball color range (obtained manually). Hence, we can filter objects with too few ball color pixels.

Shape Sieve Θ_4 : The ball object can have a shape far from a circle, but in most frames, its ratios of both width to height and height to width are less than 3, according to statistical results.

Ball Center Sieve Θ_5 : We filter objects with center colors out of the ball color range.

Each sieve Θ_i is a Boolean function on domain $\mathrm{O}(F)$.

$$\Theta_i(o) = \begin{cases} 0 & \text{if sieve } \Theta_i \text{ removes } o \in O(F), \\ 1 & \text{otherwise}. \end{cases} \quad (2)$$

After sieving, the remaining objects of $\mathrm{O}(F)$ form the ball candidate set $C(F)$ of frame F.

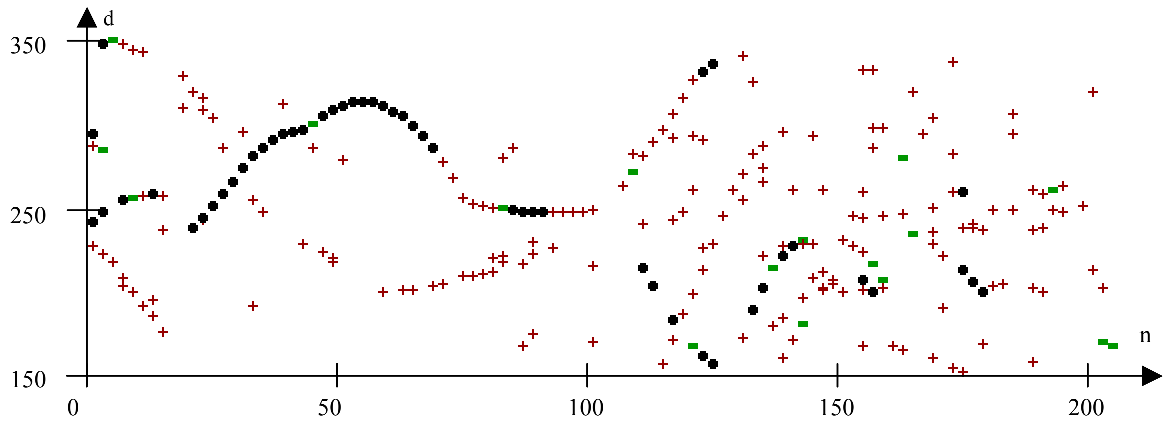
$$C(F) = \{o:\ o \in O(F),\quad \Theta_i(o) = 1 \text{ for } i = 1 \text{ to } 5.\}. \quad (3)$$

3.2.2. Candidate Classification

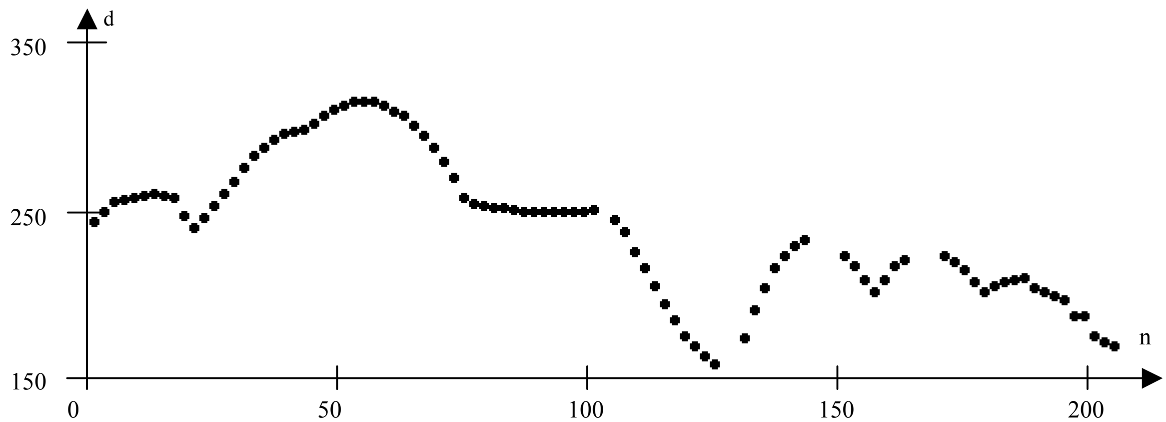
Let f_1, f_2, Λ, f_k be all features that we use to evaluate how likely a candidate is the ball. Let $o \in C(F)$ be a ball candidate of frame F and $P_i(o)$ $(i = 1, 2, \Lambda, k)$ be the probability that o is the ball regarding feature f_i. The choice of features allows us to assume that they are independent with relatively small error. This assumption is attractive because the probability that o is the ball has a simple formulae by the Bayesian rule.

$$P(\mathrm{o}) = \prod_{i=1}^{k} P_i(\mathrm{o}). \quad (4)$$

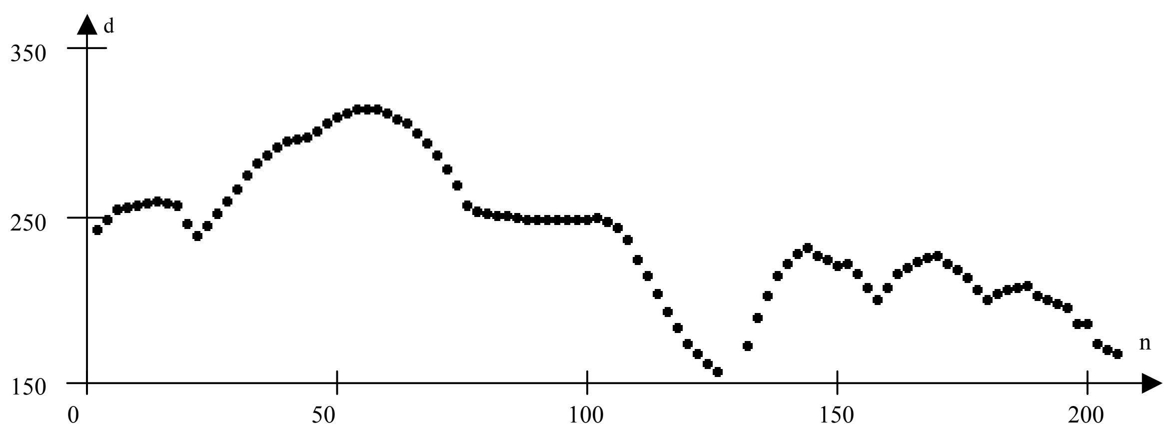
According to the probability $P(o)$, the candidates in $C(F)$ can be divided into three categories. Category 1 to 3 contains the objects with probability from high to low in order. We classify the candidates by two types of features. One is the appearance features such as the circularity, the average color distance to the ball color, and the difference to the estimated size. The other is the isolation of the candidate, i.e. how far the candidate is from other objects. This is an important feature since the candidate close to a player may be his region due to over-segmentation.



(a) The obtained candidates. Black dots, green rectangles and red crosses stand for candidates in category 1 to 3.



(b) The obtained ball trajectories after first discrimination.



(c) The obtained ball trajectories after the trajectory processing.

Figure 5. The partial candidate distance images for the different process stages for the sequence of frame 48957 to 49167 of FIFA 2002 final. In the figure, n is the serial number of the frame in the sequence and d is the distance between the candidate and the top left corner of the original frame. For the legibility, only the frames with even serial number are drawn.

3.3. Candidate Trajectory Generation

For a sequence of frames, each combination of candidate numerical features forms a CFI. The X, Y, distance, and XY images are defined in [16-17]. Figure 5(a) shows a sample distance image. We use a candidate verification procedure based on Kalman filter to find candidate trajectories in a CFI (see paper [16-17] for more detail).

3.4. Trajectory Processing

3.4.1. Trajectory Discrimination

Let $\Gamma = \{T : T \textit{ is a candidate trajectory}\}$ be the trajectory set of a given video segment in a candidate feature image. Let $\lambda_1, \lambda_2, \Lambda, \lambda_m$ be all properties of a trajectory T for evaluating whether T is a ball trajectory. A function $\Omega_i(\lambda_i)$ computes the confidential index that T is a ball trajectory with respect to λ_i. The confidential index $\Omega(T)$ to indicate how likely T is a ball trajectory is defined below:

$$\Omega(T) = \sum_{i=1}^{m} \Omega_i(\lambda_i). \qquad (5)$$

After removing the trajectory with a very small confidential index, the pseudo code in Figure 6 selects the ball trajectory from the candidate trajectories in a candidate feature image. The remaining trajectories, which are in Γ, are uncertain whether they are the ball trajectories. Both the selected ball trajectories and the remaining candidate trajectories will be extended in the trajectory extend procedure in section 3.4.2. The properties used to evaluate the candidate trajectories are the length of the longest consecutive candidate sequence of category 1 and 2, the ratios of the candidates of category 1 and 2, and the length of the trajectory.

Let Γ be the set of all candidate trajectories in a CFI.
SET the ball trajectory set B to be empty.
WHILE (Γ is not empty) **DO**
Move the trajectory T into B if it has the highest index in T and its index is larger than a given threshold.
Remove the candidate trajectories that overlap with T in Γ.

Figure 6. Ball trajectory selection procedure.

3.4.2. Trajectory Extension

For each trajectory in $\Gamma Y B$, we use a model matching procedure based on Kalman filter to locally extend it, i.e. we track the ball locally. This procedure is similar to the candidate verification procedure based on Kalman filter with the difference that the candidate verification is replaced by the model matching.

Ball Model $\mathrm{B}(c, r, l)$: It comprises three features: the color range, the ball radius upper bound, and the ball location. Let T be the trajectory to be extended and $\mathrm{B}(c, r, l)$ be the ball model. $\mathrm{B}(c, r, l)$ is initialized according to the objects in T. Once the ball in frame *k* is obtained, $\mathrm{B}(c, r, l)$ for frame *k+1* can be built. First, a Kalman filter predicts the location of the ball. A linear estimator $\mathrm{H}(\bullet)$ is used to predict the color range of the ball. In the area enclosing the predicted location, we segment the objects with color range $H(k+1)$. Then, we remove the objects with radii larger than the predicted radius upper bound $Q(k+1)$, where $Q(\bullet)$ is another linear estimator. For the detected object O, its probability being the ball is computed as below:

$$\mathrm{M}(O) = C(O) \bullet R(O) \bullet L(O). \qquad (6)$$

where $C(O)$, $R(O)$, *and* $L(O)$ are the probabilities that O is the ball with respect to color, radius, and location respectively. Finally, the object with highest probability is considered to be the ball if its probability is larger than a given threshold.

After all trajectories in $\Gamma Y B$ are extended, let $\Gamma = \Gamma Y B$. The discrimination and extension procedures work alternatively for several iterations until Γ is empty or the number of iteration reaches a threshold. This iteration is applied to several CFIs so that the selection of the ball trajectory is efficient and reliable. After we obtain all ball trajectories, we use an interpolation procedure to patch the small gaps between a pair of contiguous ball trajectories. This interpolation will obtain the ball positions when the ball is occluded by people in a short time or it is out of the camera coverage in a short time.

4. SOCCER VIDEO ANALYSIS BASED ON THE BALL TRAJECTORY

This section applies the ball trajectory computed to analyze semantic basic and complex events, team ball possession, and the play-break structure in broadcast soccer video. We propose to analyze video principally based on the ball trajectory rather than the low-level feature, which we merely use as supplementary features. The disadvantage of using only the low-level feature is that many aspects of broadcast soccer video cannot be analyzed since they are independent of the low-level feature.

4.1. Event Detection

There are two important types of interactions in the soccer game: player-ball and player-player. Player-ball interactions relate to the ball motion, whereas player-player interactions relate to the team tactics and rule offence. We detect the basic actions based on the ball trajectory. Then, the result of the basic action detection will be used as the basis to detect the more complex events.

4.1.1. Detection of Basic Actions

Here we detect two basic actions: touching and passing.

Detection of the Pivot of the Ball Motion: One of the main objectives of the players is to control the ball motion. To complete this objective, players attempt to touch the ball with the right force and direction. Each touching to the ball will alter the motion direction and velocity of the ball. Such motion change points of the ball are called the pivot points of the ball motion, or the pivots in short. Now we first detect the pivots. Let V = {*p* | *p* is a local velocity minimum point or an acceleration start point}. Let $f_1(p) = r$ and $f_2(p) = c$ be the functions, representing Y curve and X curve of the ball position over frames, where *r* and *c* are the row and the column of the ball center in the frame *p* respectively. Let $S_1 = \{p \mid p$ is a local trajectory maximum of $f_1\}$, $S_2 = \{p \mid p$ is a local trajectory minimum of f_1}, and S_3 = {p | p is a local trajectory maximum or minimum of f_2}. Then, we consider $S=V+S_2+S_3-S_1$ to be the pivot set of the segment. Figure 7 shows the pivot detection result for a sample segment in which a vertical bar indicates a pivot point. In Figure 7 to 10, v stands for the ball velocity and n stands for the serial number of the frame.

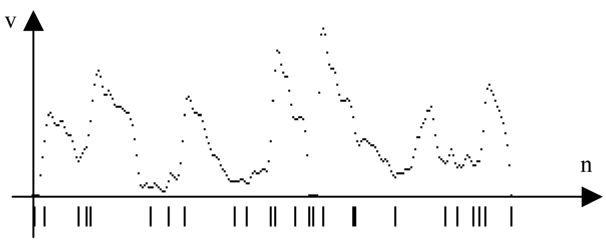


Figure 7. Pivots from ball trajectory (vertical bars).

Touching Detection: A touching point is a frame where a person touches the ball. The touch includes the ball kick by foot, and other categories of touchings. When a person touches the ball, the ball curve should form a pivot point. On the other hand, the pivots can also be formed by other factors, such as camera motion, ball bounce, etc. Thus, for each pivot we check whether a person touches the ball. The pivots where nobody touches the ball are removed, and the rest form a set of touching points. Figure 8 shows the curve of the ball velocity over frames in which the black vertical bars indicate the touching points.

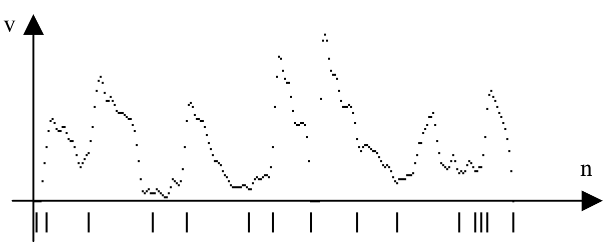


Figure 8. Touching points (vertical bars).

Detection of Passing: A passing is an action that a player passes the ball to his teammate. Normally, such an action produces a significant ball trajectory. Passings are valuable for understanding the game as they consist of large portion of the game. So we design a scheme to detect the passing. Besides the passing, the soccer game comprises mainly fighting, dribbling, possession transit, and shooting. These actions consist of smaller portion of the game. Figure 9 shows the obtained passings for a sample video segment, in which a horizontal line segment with two vertical bars at its two ends indicates a passing.

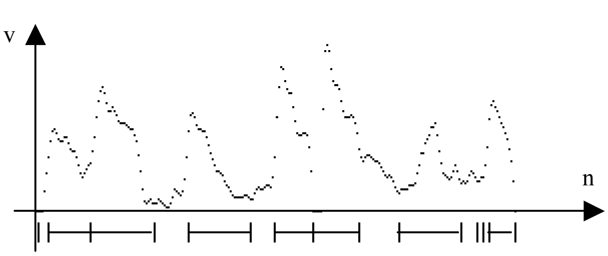


Figure 9. Passings (line segments between two bars).

4.1.2. Detection of Complex Events

Complex soccer events are detected based on the detection result of the basic actions, the ball trajectory, and the result of mark detection. We choose goal as an example to show how the complex events are detected in our proposed approach. The steps involved in goal detection are shown in Figure 10. In the first phase, we detect the goalmouth, and then we detect the shooting against the goalmouth. In the second phase, we further find the ball that is close to the goalmouth for a number of frames. However, the fact that the ball is in the goalmouth in a frame can not conclude that there is a goal since we are processing broadcast soccer video. Hence, we decide whether there is a goal by considering the ball trajectory and goalmouth position relation. Notice that in a just-missing shooting, the ball might also be in the goalmouth for a few frames. Indeed, the challenge we are facing is how to avoid classifying the just-missing into the goal. We can further verify the goal with whistling and audience shout.

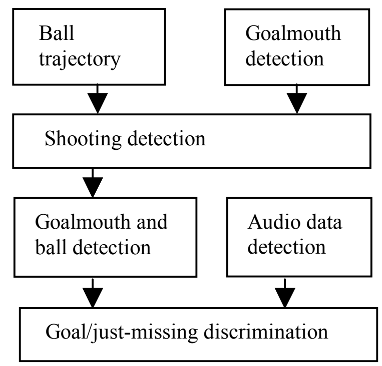


Figure 10. Architecture of goal detection.

4.2. Team Ball Possession Analysis

We analyze the team ball possession based on the result of touching detection as described above. To credit the ball possession, we must determine which team touches the ball. For each frame with a touching, the objects in the ball neighborhood are extracted first. Then, each object is evaluated whether it is a person. Thus, people in the ball neighborhood are extracted. Last but not least, a Support Vector Machine is used to recognize the team of each person. In addition, when a team consecutively touches the ball, we consider that the team possesses the ball.

4.2.1. People Extraction

We use a seed-growing procedure to find the objects that are in the ball neighborhood in the field before removing non-person objects. Let f_1, f_2, Λ, f_k be all the features for evaluating how likely an object is a person. Assume that $K(F)$ be the set of all detected objects in the ball neighborhood of frame F. Let $o \in K(F)$ be an object and $P_i(o)$ $(i = 1, 2, \Lambda, k)$ be the probability that indicates how likely o is a person with respect to the feature f_i. The choice of features allows us to assume that they are independent with relatively small error. With this assumption, the probability that o is a person has a simple formulae by Bayesian rule.

$$P(o) = \prod_{i=1}^{k} P_i(o). \quad (7)$$

The object o is removed if its probability $P(o)$ is too small; otherwise we further identify its team in the section 4.2.2. We use the size as main feature to evaluate whether an object is a person since we have already known the size of people from the ball. We divide the object into two objects if its size is larger than the size of a person. Another property is the ratio of the width to height. Final property is the ratio of its area to the area of its bounding box.

4.2.2. Team Recognition

In a soccer game, the people on the field belong to five categories: the players in Team A and B, goalkeepers in Team A and B, and the referee. The shirts of these five categories of people are in five different colors. Hence, the color histogram of a person can determine their type. For each type of people, we manually identify a color that differentiates them from other people in advance. For each such color, we build several color bins which span a range of color around it. Then for each person, we calculate the distribution of pixels with colors falling into the pre-built bins. This distribution forms a color histogram, which is used to recognize his team through a Support Vector Machine [2].

4.3. Play-Break Structure Analysis

In a soccer game, the ball is either in play or break [13-14]. There is a break when the ball is out of the field or the referee stops the game. In the soccer video, almost all play frames have the ball, but not all the frames with the ball are play frames. Thus, the ball trajectory provides the solid basis for the play-break structure analysis. A series of the connected trajectories form a ball curve, whose two ends may not be in play. Thus, our work is to decide how long of two ends are not the play portion. Besides the motion velocity of the ball, whistling is another reference for determining play-break cutting point. Hence, we analyze the play-break structure based on the ball trajectory with the aid of whistling. Figure 11 depicts the architecture of play-break analysis.

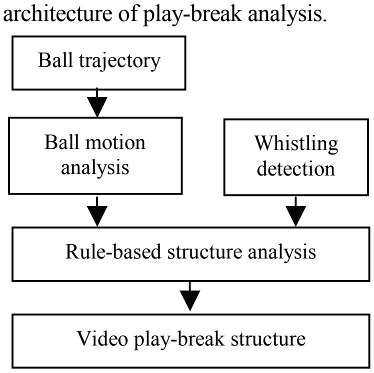


Figure 11. Architecture of play-break analysis

4.3.1. Whistling Detection

In soccer game, whistling has three types: long, double, and multiple. However, all whistlings indicate game resume and stop, i.e. they are good reference points for play-break analysis. We have developed an automatic whistling detection approach for the soccer video [12]. In the system of analyzing play-break structure, we consider the result of whistling detection as one of the inputs.

4.3.2. Structure Analysis

Now we describe the functions of the system of play-break analysis shown in Figure 11. The inputs are the ball trajectory and the whistling over frames (time). In the play period, the ball moves fast. Hence, the ball velocity is another good indicator whether the frame belongs to the play portion too. The primary goal of the first phase of the system is to compute the derived information such as the ball motion velocity and orientation that are required by the rule-based procedure. The rule-based procedure decides the separation points by fusing the derived information from the ball trajectory and the whistling. Figure 12 shows the result for finding one play/break separation point.

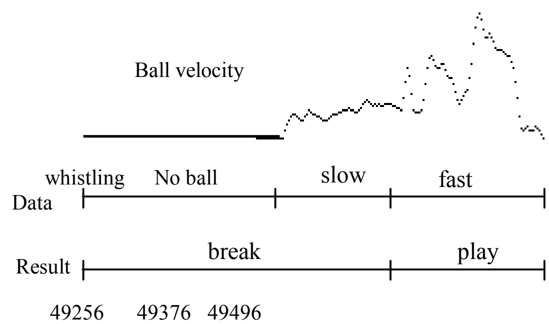


Figure 12. A play/break separation detection result for FIFA 2002 final (frame 49256 to 49496).

5. EXPERIMENTAL RESULTS

5.1. Results of Finding the Ball

Before applying the ball detection and tracking algorithm, we have a procedure to automatically find the video segments with soccer field. The test has been conducted on 15 sequences (176 seconds in total) with the field. The content of the video used is the final match of 2002 FIFA World Cup Final. These sequences are representative in the way that they include short to long sequences, ball-less sequences, and sequences with assorted frames of closed-up and full view frames. In Table 1, the detected balls are the balls that are selected by the first discrimination. A frame is said to be with the ball if people can conclude the existence of the ball in the frame based on the previous and posterior frames even though the ball may be occluded but it must be in the field. Frames are considered to be no ball if a player holds the ball for a long time before starting the ball. A frame is said to be detected correctly if the algorithm can conclude correctly the existence of the ball and the correct position. However, our ball candidate detection procedure does obtain a lot of false candidates since these false candidates resemble the ball as shown in Figure 5(a). Fortunately, the ball trajectory processing procedure can identify the ball trajectories correctly as shown in Figure 5(b). A ball trajectory might have some non-ball objects---false positive detection, but these non-ball objects are very close to

Table 1. The detection and tracking results for 15 consecutive sequences of frames.

Sequences with The soccer field	Ground truth			Detection result			Detection and tracking result		
	# frame	# ball	# no ball	# detect	# false positive	accuracy	# det. & tracked	# false positive	accuracy
002900-003001	102	102	0	98	0	96.1%	102	0	100%
003143-003308	266	108	158	266	0	100%	266	0	100%
005368-005503	136	130	6	113	0	83.1%	136	0	100%
005623-005834	212	212	0	196	0	92.5%	212	0	100%
008026-008424	399	323	76	281	3	70.0%	385	3	96.1%
008805-008834	30	0	30	30	0	100%	30	0	100%
008950-009069	110	0	110	110	0	100%	110	0	100%
048957-049167	211	211	0	192	3	91.0%	209	3	99.1%
049256-049974	719	608	111	678	2	94.3%	702	4	97.6%
Six sequences	2218	1652	566	1958	13	88.3%	2131	15	96.1%

the real balls. The false positive detection may only appear when a player with a pair of white shoes or socks is dribbling the ball. Fortunately, they do not misguide the tracking procedure since they are very close to the real balls. The ball trajectory discrimination does not obtain any non-ball trajectory. Figure 5(c) shows the ball trajectories after trajectory processing. Figure 5(a-c) shows the candidate distance images for the same sequence in the different process stages. However, some ball trajectories cannot be found since they are too short. The detection and tracking results for the 15 representative sequences are shown in Table 1.

5.2. Results of Video Analysis

For semantic analysis to broadcast soccer video, the inputs to our detection modules are the ball trajectory and the video. The various analysis results are shown in Table 2-4.

5.2.1. Results of Event Detection

We can detect almost all the touchings appearing in the frames. Unfortunately, not all touchings appear in the frame. Furthermore, when the touching is not shown in the frame, we cannot know whether the corresponding trajectory is a pass. This is the main cause that we are unable to detect some passes. Initially, we have tested with 6 broadcast soccer segments that contain 27 touchings and 18 passings for touching and passing detections. The total length of the 6 segments is 55 seconds. For goal detection, we have tested with 2 games which occurred in day and night respectively. Two games contain 5 goals and 16 just-missings. For goal and just-missing detection, we differentiate them by checking the ball trajectory and goalmouth relation. However, the issue of this discrimination is that for some just-missings, the ball is very close to the goalmouth in a number of frames. Thus, the algorithm considers them as goals.

Table 2. Event detection performance.

Event	Precision	Recall	# events
Touching	95.6%	81.5%	27
Passing	100%	77.8%	18
Goal	100%	100%	5
just-missing	81.2%	100%	16

5.2.2. Results of Team Ball Possession Analysis

We cut the play part of the soccer video at its each touching point. Hence, the play part is divided into a set of touching segments. A touching segment is called a Team A (B) possession segment if its two touchings are from Team A (B). For the team possession analysis, the player team discrimination is reliable and the error mainly inherits from the error of touching point detection. In the expression "v/w" in Table 3, v is the correctly detected number and w is the ground truth of team ball possession.

Table 3. Team possession analysis performance**.**

Sequence	# fr.	Brazil	Germ.	Accur.
002900-003001	102	100/80	0/0	78.4%
003240-003308	169	0/0	169/169	100%
005368-005503	136	0/0	0/0	100%
008026-008296	271	183/271	0/0	67.5%
048957-049102	146	0/0	100/146	68.5%
049415-049974	560	0/0	481/560	85.9%

Table 4. Play-break analysis performance.

ground truth			Results	
Sequence	# frame	# play	# cor't.	accur.
002900-003001	102	102	80	78.4%
003143-003308	266	68	221	83.1%
005368-005503	136	136	136	100%
005623-005834	212	0	212	100%
008026-008424	399	270	399	100%
008805-008834	30	0	30	100%
008950-009069	110	0	110	100%
048957-049167	211	211	211	100%
049256-049974	719	560	719	100%

5.2.3. Results of Play-Break Analysis

Our play-break analysis results are accurate since most frames with the ball are the play frame. The mistake is mainly from the cases when there are the very short breaks without whistling and the camera does not move much.

6. CONCLUSIONS AND FUTURE WORK

We have presented a trajectory-based algorithm, which is improved from the algorithm given in [16-17], for detecting and tracking the ball in broadcast soccer video. Compared with the algorithm presented in [16-17], we used an iterative procedure, which is more reliable than the previous naïve procedure, to process the candidate trajectory. Moreover, we added goalmouth and ellipse detection to improve the ball size estimation. The experimental results show that the improved algorithm boosts the accuracy of the ball detection and tracking from about 85% to above 96% for the video segments in which all frames without the field have been removed in advance. Consequently, the results of finding the ball are much better than the existing results in [3, 6, 15] even though the videos used in [3] are not from TV signal. The presented algorithm contains four contributions compared with the object-based algorithms. The first contribution is the ball size estimation for each frame by using the sizes of certain salient objects as the input. The second contribution is ball candidate detection for each frame. These ball candidates contain sufficient data to detect the ball trajectories while all the candidates for a long sequence have only a small data volume. The third contribution is the candidate feature images that present the spatial and temporal data together. Thus, these images allow the temporal filters to be applied to them. The fourth contribution is trajectory analysis. Kalman filter effectively obtains candidate trajectories in candidate feature images. The trajectory discrimination procedure can distinguish the ball trajectories from other trajectories. Furthermore, the trajectory extension procedure tracks the ball to extend the ball trajectories and the remaining candidate trajectories. In addition, both trajectory discrimination and extension procedures work iteratively, so that the algorithm can identify the ball trajectory very reliably.

In ball detection and tracking, there are several avenues for the future work. (a) With the facility of the ball trajectory, we have analyzed the team ball possession. However, we can further analyze the ball possession for a particular player. (b) The trajectory-based method presented in this paper may solve other ball detection and tracking problem such as football, tennis, golf tracking, and basketball. (c) This method may also solve some other object tracking problems.

The novelty of the proposed soccer video analysis method is that it is based on the ball trajectory rather than the low-level feature. In this approach, it detected pivots, touching, and passing that cannot be detected by the existing approaches. Furthermore, this paper analyzed the team ball possession, which is another analysis which is impossible to be performed by the existing approaches. In addition, we improved the play-break analysis and goal detection. To analyze the video, we build the rule-based model for each game event which derives from the soccer game rules rather than heuristics. The ball trajectory plays an important role in building models as the ball motion has a direct relation with game events. Two basic actions are detected in the first phase of the event detection module based on the ball trajectory. A more complex event--goal, is detected in the second stage based on the detected basic actions and the results of mark detection. Our primary results are very promising.

In the soccer video analysis, future work includes improving our method to detect more events and more details of the events. In play-break analysis and goal detection, we will compare our algorithms with the existing algorithms.

7. ACKNOWLEDGEMENT

Thanks to Prof. Shih-Fu Chang, Dr. Yihong Gong, Dr. Jesse Jin, and the anonymous reviewers for their inspiring comments and encouragement.

8. REFERENCES

[1] S. F. Chang. The Holy Grail of content-based media analysis, IEEE Multimedia, vol. 9:2(2002), 6-10.

[2] C. Cortes and V. Vapnik, Support-Vector Networks, *Machine Learning*, Vol. 20(1995), 273-297.

[3] T. D'Orazio, N. Ancona, G. Cicirelli, and M. Nitti. A ball detection algorithm for real soccer image sequences, In Proc. ICPR, Canada, Aug. 11-15, 2002

[4] A. Ekin, A. M. Tekalp. Robust dominant color region detection with applications to sports video, submitted to *Comp. Vision & Image Understanding*.

[5] A. Ekin, A.M. Tekalp, and R. Mehrotra. Automatic Soccer Video Analysis and Summarization, *IEEE Trans. on Image Processing,* Vol. 12:7(2003), 796-807.

[6] Y. Gong, T. S. Lim, H. C. Chua, H. J. Zhang, and M. Sakauchi. Automatic Parsing of TV Soccer Programs, 2nd Int. C. on Multimedia Comp. and Sys. , 1995, 167-174.

[7] M. Isard and A. Blake. ICondensation: Unifying low level and high level tracking in a stochastic framework. In Proc. of European Conf. on Comp. Vision, 1998, 767--781.

[8] G. Kuhne, S. Richter, and M. Berer. Motion-based segmentation and contour-based classification of video objects, In Proc. ACM MM, Canada, 2001.

[9] D. Lowe. Robust model based motion tracking through the integration of search and estimation. Int. J. on Comp. Vision, Vol. 8(1992), 113-122.

[10] C. Rasmussen and G. Hager. Joint probabilistic techniques for tracking multipart objects. In Proc. CVPR, 1998, 16-21.

[11] V. Tovinkere and R. J. Qian. Detecting semantic events in soccer games: Towards a complete solution, In Proc. ICME 2001, 1040-1043.

[12] M. Xu, N. C. Maddage, C. Xu, M. Kankanhalli, and Q. Tian. Creating audio keywords for event detection in soccer video, In Proc. ICME 2003, Vol II, 281-284.

[13] L. Xie, S. F. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with hidden markov models, In Proc. ICASSP 2002.

[14] P. Xu. L. Xie, S. F. Chang, A. Divakaran, and H. Sun. Algorithm and system for segmentation and structure analysis in soccer video, In Proc. ICME 2001.

[15] D. Yow. B. L. Yeo, M. Yeung, and B. Liu. Analysis and presentation of soccer highlights from digital video, In proc. ACCV, 1995.

[16] X. Yu, Q. Tian, and K. W. Wan. A novel ball detection framework for real soccer video, In Proc. ICME 2003, Vol II, 265-268.

[17] X. Yu, C. Xu, Q. Tian, and H. W. Leong. A ball tracking framework for broadcast soccer video, In Proc. ICME 2003, Vol II, 273-276.