Estimation of Ball Route under Overlapping with Players and Lines in Soccer Video Image Sequence

Takumi Shimawaki, Takuro Sakiyama, Jun Miura, and Yoshiaki Shirai
Dept. of Mechanical Engineering, Osaka University
2-1, Yamadaoka, Suita, Osaka 565-0871, Japan
E-mail:{simawaki, saki, jun, shirai}@cv.mech.eng.osaka-u.ac.jp

Abstract

This paper deals with the analysis of broadcast soccer video. To recognize interesting events such as a goal, estimation of ball movements is necessary. It is, however, sometimes difficult to detect a ball by a simple color and shape-based method when it overlaps with players and lines. We therefore develop a method of estimating a ball route during such overlaps by considering spatio-temporal relationships between players, lines, and the ball. The method can deal with difficult cases such as the one where a ball disappears at a player and re-appears from another player. Experimental results show the effectiveness of the method.

1 Introduction

There is an increasing demand for summarizing a video image sequence of a soccer game (or other sports) into a set of interesting scenes such as goal scenes so that viewers can quickly survey the game. We are now developing a system for retrieving interesting and informative scenes based on scene understanding. To understand various scenes in soccer games, it is essential to know the movements of players and a ball. This paper focuses on ball tracking.

Previous ball detection methods use, for example, SVM [1] or a generalized Hough transform [2]. Most of the previous methods are, however, applicable only when a ball is sufficiently large and not so fast in the image. In our case, a ball is usually small and sometimes moves fast. Moreover, it sometimes overlaps with players and lines.

Several ball tracking methods deal with such problems by applying statistical filters such as Kalman filter [3] and particle filter [4] or by using ball trajectory models [5, 6]; these methods can handle short-term occlusions or overlaps. In actual scenes, however, players and lines often overlap with each other and, as a result, a ball which has overlapped with (or occluded by) a player may appear from another player's region. To treat such cases, it is necessary to examine possible *routes of the ball* (i.e., a sequence of objects which overlap with (or occlude) the ball) based on spatio-temporal relationships between players, lines, and the ball.

(a) player (b) line (c) the stands

Figure 1. Example cases where a simple ball detection fails.

This paper proposes a method of estimating the route of a ball when it overlaps with players and lines in broadcast soccer video. We focus on the images taken from the center camera which spans the widest area of the field. We automatically detect a shot from that camera and estimate the initial camera parameters at the first frame of the shot from the correspondence between the detected lines and the model of the field [8]. We then on-line estimate the camera parameters in subsequent frames by line tracking [7].

2 Overview of the method

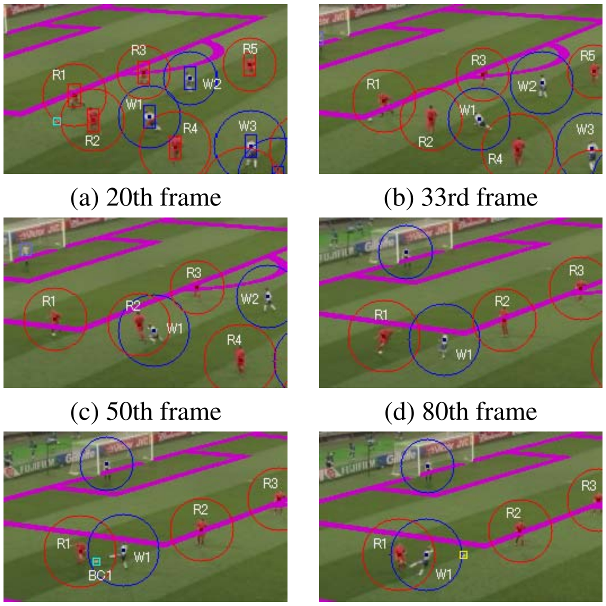
We consider the following three cases where the ball cannot be detected using a simple ball detector (see Fig. 1).

1. Overlaps with players (or referees) (see Fig. 1(a)).
2. Overlaps with lines (see Fig. 1(b)).
3. Overlaps with the stands (see Fig. 1(c)).

Difficult situations arise when the ball continuously overlaps with several players and lines. Fig. 2 shows such a situation: (a) a ball is detected near a white player (W1); (b) the ball overlaps with a line; (c)-(d) a red player (R1) keeps the ball; (e) he kicks the ball; (f) the ball is detected again at the back of the white player (W1). Although the ball disappears and re-appears near the same player (W1), it takes a complicated route during this overlapping period.

We examine the frames between the disappearance and the re-appearance of the ball to estimate the ball route in the following steps.

Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE



(a) 20th frame (b) 33rd frame

(c) 50th frame (d) 80th frame

(e) 87th frame (f) 93rd frame

Figure 2. An example scene where a ball has not been detected for a long period.

1. Enumerate possible transitions of the ball between objects (players, lines, or the stands) that overlap with the ball.
2. Generate ball route candidates considering spatio-temporal relationships between the objects and the ball.
3. Generate a rough ball trajectory for each ball route candidate, if possible, by considering the constraints on ball movements.
4. Evaluate the trajectories based on the detection of ball-like objects on them, and select the best trajectory and thus the best ball route.

3 Enumerate ball transitions

We construct a graph called a *transition graph* that enumerates possible transitions of a ball between objects. Nodes of the graph consist of objects that may overlap with the ball, that is, players, lines, and the stands. A *ball candidate*, which is an isolated ball candidate, is also represented as a node. Links consist of possible transitions between the nodes. Lines on the ground are divided into straight lines and curved ones; the stands are divided into four regions. Fig. 3 shows the nodes of lines and the stands.

We consider the following transitions between objects:

1. player ⟷ player, ball candidate, line.
2. ball candidate ⟷ line.
3. line or the stands ⟷ line or the stands

The transitions including players and ball candidates are *temporal* and effective only while two nodes are close

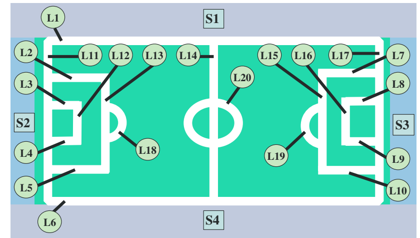


Figure 3. Division of lines and the stands

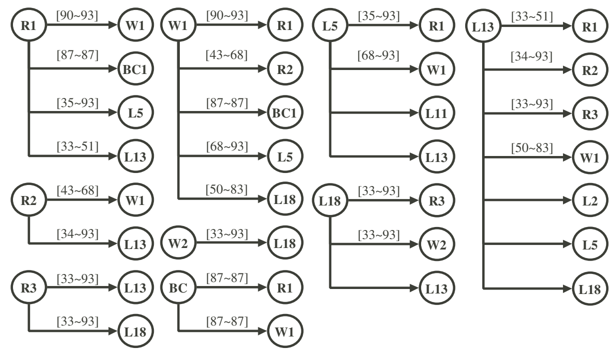


Figure 4. Part of transition between nodes from Fig. 2

enough. In Fig. 2, the circle drawn around each player shows the range within which the ball can move in the next frame. If there is another player's centroid or a line in this circle, the ball may make a transition to one of them. Transition between lines and the stands are *fixed* and those between adjacent nodes in Fig. 3 are possible.

Fig. 4 shows a part of transitions generated from Fig. 2. Labels of nodes, W*, R*, L*, BC* indicate white players, red players, lines, and ball candidates, respectively. For example, since a white player (W1) exists near a red player (R1) during frames 90–93 (see Fig. 2(f)), transition R1→W1 for that period is generated; also, since R1 is near a line (L5) during frames 35–93 (see Fig. 2(c)-(e)), transition R1→L5 is generated.

4 Generate ball route candidates

We generate ball route candidates by searching the transition graph for possible routes connecting the node where a ball disappears and the node where the ball re-appears. In this process, we consider the temporal consistency of transitions. That is, the earliest frame of the transition that gets into a node should be earlier than the latest time of the transition that gets out of the node. We also use the following rules to avoid generation of unrealistic transitions.

1. One player node can appear only once in a route. This is to avoid ball movements in which a ball moves back and forth between the same players.

Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE

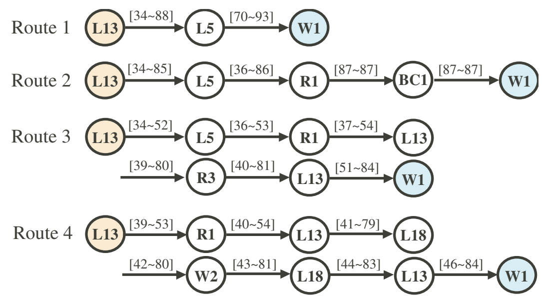


Figure 5. Some of generated ball route candidates from Fig. 4

2. The maximum number of successive line nodes is two. In addition, the shape of two successive lines should be a physically-possible ball movement. In Fig. 2(b), for example, the transition R1→L13→L18→W2 is possible because the lines L13 and L18 can be approximated by a parabola in the image and the ball movement can sometimes be parabolic.
3. The maximum number of successive stands nodes is two.

Fig. 5 show some of the candidates generated from Fig. 4.

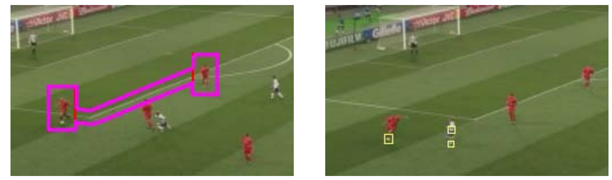
5 Generate and evaluate rough ball trajectories

Ball detection by a simple ball detector during an overlapping period is difficult. Nevertheless we would like to have evidence used for ranking ball route candidates. Therefore, we first generate rough ball trajectories from a set of *ball-like regions*, which cannot be detected by the simple detector but match with a ball model to some extent, and then evaluate them to select the most probable ball route.

5.1 Extraction of ball-like regions

We search the area determined by each ball route candidate for ball-like regions with *separability filter* [9]. Seperability filter responses to cocentric circular patterns and is often used for detecting eyes in face recognition.

The regions that have higher filter responses than some threshold are extracted. Since white players' shirts and socks regions output high responses, we try to remove them as many as possible. We remove shirts regions using the result of player tracking. We remove socks regions by examining their shape; if the ratio of the longer principal axis to the shorter one of a region is larger than a threshold, the region is considered to be a socks region. Fig.6(b) shows the ball-like regions inside a part of the search region. Since several white regions other than the ball region may remain as shown in Fig. 6(b), we use the motion continuity to filter out such regions, as described below.



(a) search area (b) result

Figure 6. Detection of ball-like regions

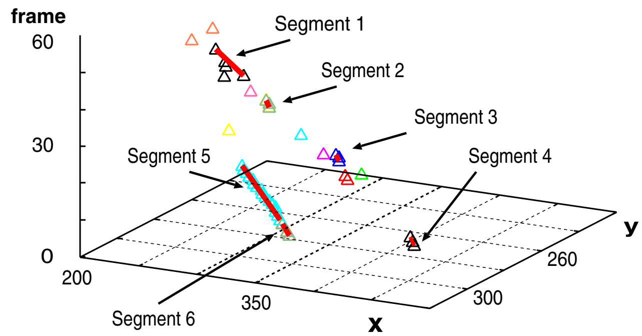


Figure 7. Clustering of ball-like regions and segment extraction

5.2 Generation of sequences of ball-like regions

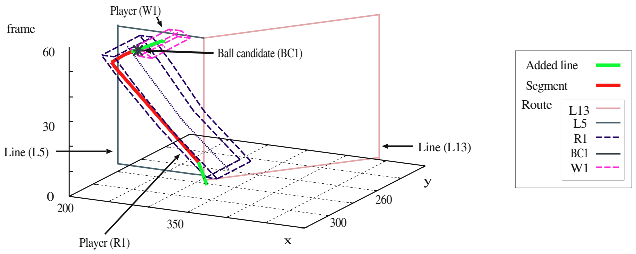
We generate sequences of ball-like regions (called *segments*). We first perform a simple clustering of the regions; if two regions are within a certain distance *in space and time*, they are put in a cluster. Clusters with less than three regions are deleted. We then fit a line to each cluster to generate a segment. Fig. 7 shows a result of clustering and segment generation for Route 3 in Fig. 5. Triangles are extracted ball-like regions and their colors indicate cluster IDs; lines indicate generated segments.

The sum of the output of the separability filter for the regions in a segment is called the *score* of the segment and is used for selecting the most probable ball route.

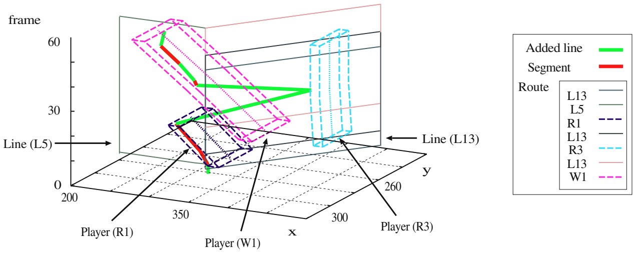
5.3 Generation of rough ball trajectories

We generate a set of rough ball trajectories during the overlapping period; a trajectory should pass all nodes of the ball route candidate under consideration and include at least one segment. In rough trajectory generation, we first enumerate all possible combinations of segments such that no more than one segment exist at a time. We then try to generate a trajectory for each combination by adding straight lines to connect the segment in the combination. The lines should satisfy the constraint of the maximum ball speed.

Fig. 8 shows two rough ball trajectories generated from the segments shown in Fig. 7, for Routes 2 and 3 in Fig. 5. Each trajectory is generated so that it passes the nodes (players, lines, and ball candidates) in the corresponding routes. Red bold lines and green ones indicate the segments



(a) Trajectory for Route 2 in Fig. 5.



(b) Trajectory for Route 3 in Fig. 5.

Figure 8. Examples of generated trajectories.

(a) for Fig. 8(a) (b) for Fig. 8(b)

Figure 9. Ball movements projected onto the maosaicked image.

(i.e., sequences of extracted ball-like regions) and the added lines, respectively. The existence of players and lines in space-time is also shown. Fig. 9 illustrates how the ball moves in the mosaicked image in the case of Route 2 and 3.

5.4 Selection of Most Probable Ball Route

The score of a feasible trajectory is the sum of the scores of its segments normalized by the total length of the trajectory. The score of a ball route candidate is calculated as the highest among the scores of the corresponding feasible trajectories, and the ball route with the highest score is selected. In the above example, the scores for Route 2 and 3 are 22.24 and 4.00, respectively, and Route 2 (L13→L5→R1→BC1→W1) is finally selected.

5.5 Results for Other Sequences

We tested the method for a long soccer video of about nine and a half minutes. In the video, 2,289 frames are in the shots from the center camera. In these frames, there are 34 sequences during which a ball is not detected by a simple detector and to which the proposed method is thus applied. Their average number of frames is about 20. We examined the outputs of the method for these sequences and found the estimated route is correct for 28 sequences. Failures in ball route estimation are mostly caused by those in detecting ball-like regions, especially when a ball goes out of the field into the stands immediately after a player kicks or heads it.

6 Summary

This paper has presented a method of estimating a ball route in soccer broadcast video when a ball continuously overlaps with players and lines. We first generate a transition graph representing possible transitions of the ball between overlapping objects, based on their spatio-temporal relationships. We then enumerate ball route candidates from the graph and select the best one by searching for the evidence for ball existence near each route candidate. By this two-stage approach, we can greatly reduce the region to examine in the image.

The current method estimates a ball route *in the image*; when a ball passes a player, for example, the method does not tell whether or not the ball is actually touched by the player. A future work is to perform additional inference about the movement of the ball and players in the scene. Another future work is to apply the proposed method to the shots other than the ones from the center camera.

References

[1] N. Ancona et al. Ball Detection in Static Images with Support Vector Machines for Classification. *Image and Vision Computing*, Vol. 21, pp. 675–692, 2003.

[2] T. D'Orazio et al. A New Algorithm for Ball Recognition Using Circle Hough Transform and Neural Classifier. *Pattern Recognition*, Vol. 37, pp. 393–408, 2004.

[3] X. Yu et al. A Ball Tracking Framework for Broadcast Soccer Video. *Proc. of Int. Conf. on Multimedia and Expo*, Vol. 2, pp. 273–276, 2003.

[4] K. Choi and Y. Seo. Probabilistic Tracking of the Soccer Ball. *Proc. of ECCV Workshop on Statistical Methods in Multi-Image and Video Processing*, pp. 50–60, 2004.

[5] Y. Ohno, J. Miura, and Y. Shirai. Tracking Players and Estimation of the 3D Position of a Ball in Soccer Games. *Proc. ICPR-2000*, pp. 145-148, 2000.

[6] J. Ren et al. A General Framework for 3D Soccer Ball Estimation and Tracking. *Proc. IEEE Int. Conf. on Image Processing*, 2004.

[7] A. Yamada, Y. Shirai, and J. Miura. Tracking players and a ball in video image sequence and estimating camera parameters for 3d interpretation of soccer games. *Proc. ICPR-2002*, pp. 303–306, 2002.

[8] T. Shimawaki et al. Analyzing Video Image Sequence for Scene Retrieval System in Soccer Games. *Proc. of MIRU-2005*, pp. 1129-1136, 2005 (in Japanese).

[9] K. Fukui and O. Yamaguchi. Facial Feature Point Extraction Method Based on Combination of Shape Extraction and Pattern Matching. *Systems and Computers in Japan*, Vol. 29, No. 6, pp. 696–710, 1999.