Where are the ball and players?
: Soccer Game Analysis with Color-based Tracking and Image Mosaick

Sunghoon Choi, Yongduek Seo, Hyunwoo Kim, Ki-Sang Hong
Dept. of EE, Pohang University of Science and Technology
San 31 Hyoja Dong, Pohang, 790-784, Republic of Korea
e-mail : {shoon, dragon, pccc, hongks}@indy.postech.ac.kr

Abstract

Knowing the locations of players and the ball in a ground field is the key for soccer game analysis, which is the main subject of this paper.

Given an image sequence, we address here three main problems: 1) ground field extraction, 2) player and ball tracking and team identification and 3) player positioning. The region of ground field is extracted on the basis of color information, to which all the other processing is restricted. Players are tracked by template matching and Kalman filtering. Occlusion reasoning is done by color histogram back-projection. To find the location of a player, a field model is constructed and a transformation between the input image and the field model is computed using feature points when the center circle is visible. Otherwise, image-based mosaicking technique is applied. By this image-to-model transformation, the absolute position and the whole trajectory on the field model is determined.

We tested our method on real image sequence and the experimental results are given.

1 Introduction

In this work we are to solve the problem of tracking players and the ball in the image sequence of a soccer game which is captured from a broadcasted TV signal. Because of inter-field motion difference, we use only even field of a frame. Sample images of the sequence are shown in Figure 1. Our task is to track all players and the ball presented in the sequence and to find

Figure 1: Two 640x240 images(fields) of a soccer play sequence. The video is sampled at thirty frames per second and deinteraced

their positions on the ground field. Three main problems for automatic soccer game analysis are addressed here.

1. The ground field should be extracted in order to track players, to find the half line, a side line and the center circle, and to make mosaic image for computing image-to-model transformation. Because the result is used in all the following procedures, it should be as accurate as possible. We used a color histogram information under the assumption that the region of the ground is nearly green and occupies almost areas of images

2. Each players and the ball should be identified and tracked in the image sequence. Players move non-rigidly, frequently collide each other and are occluded by other ones. Template matching and Kalman filtering is applied for player tracking. Occlusion reasoning is done by color histogram back-projection method[7].

   Players of the same team is grouped together (team identification). This is done by comparing spatial color distribution of player templates. A similar method is used in ball tracking. However, since the ball is too small to track alone, a method is devised to reason about the location of it.

3. The absolute positions of the players should be known. Because there are small number of features in soccer sequence, we can not easily find the absolute locations of players on the field. A field model is constructed and image-to-model transformations are computed to attack the problem. When the center circle of the field model is found in an image, the transformation is computed by the locations of four feature points from the image of the center circle. Otherwise image mosaicking technique [3] is used to find out the image-to-model transformation. Then the trajectory of each player is computed. These trajectories show moving pattern of a player or a group of players. Also it can be used in game analysis after the game or video annotation.

A comparable work to ours is that of Intille and Bobick [4] in which a method is proposed to track players in the images of American football game. There are lots of yard lines and other features in their images which help to compute the image-to-model transformation. In our case, however, only small number of features are visible, which makes it difficult to compute the image-to-model transformation. When players are occluded, they select distinctive features based upon the objects in the closed-world (ground field) and track them. We use the color histogram back-projection algorithm to solve the occlusion problem. Kawashima[6] tried to analyze the group behavior of soccer players using color histogram projection method. Taki[8] developed a motion analysis system to evaluate teamwork qualitatively in soccer games given images from multiple cameras. Yow[9] proposed a method to show a big panoramic highlight scene by applying mosaicking technique to soccer image sequences. These systems mainly aimed at obtaining qualitative information or just presenting a special view. On the contrary, our system track each player and the ball, identify the player's team and compute the absolute location of the players.

In section 2 field extraction method is presented. Player and ball tracking method and team identification are in section 3, and the field model and the method for computing absolute player position in section 4. Finally conclusion remarks and future research topics are given in section 5.

2 Field Extraction

Because all the players are on the field, we extract the region of the ground field first by segmenting out non-field regions like ad.'s, and then the players are extracted on this field on the basis of the field extraction. It is assumed that the field has a uniform color of green and occupies large area in the image. So we calculate histogram of each color, R/G/B, and find peak values R_{peak}, G_{peak}, B_{peak}, respectively. Figure 2 shows an example of the color histogram and the peak values. Because the field occupies the largest area in the image, we can assume this color represents the field. Using these peak values, input image is binarized.

$$B(x,y) = \begin{cases} 1 & : \begin{array}{c} |I_R(x,y) - R_{peak}| < R_{th} \\ |I_G(x,y) - G_{peak}| < G_{th} \\ |I_B(x,y) - B_{peak}| < B_{th} \\ I_G(x,y) > I_R(x,y) \\ I_G(x,y) > I_B(x,y) \end{array} \\ 0 & : \text{otherwise} \end{cases}$$

where, $B(x,y)$: binary image, $I_R(x,y)$, $I_G(x,y)$, $I_B(x,y)$: R/G/B images, R_{th}, G_{th}, B_{th} : threshold values for R/G/B.

After making binary image $B(x,y)$, morphological filtering is applied to $B(x,y)$, and the largest

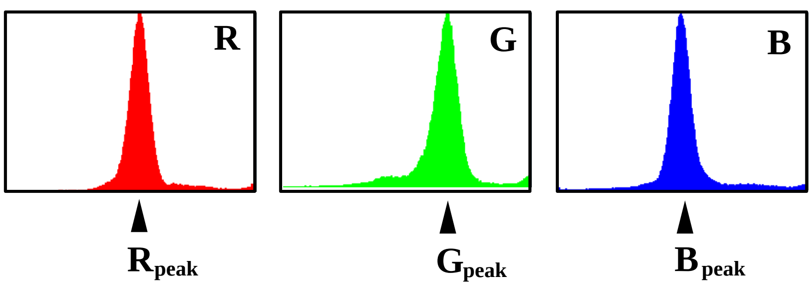


Figure 2: Global histogram of an input image

connected component is extracted that represents the field but has many holes caused by colors of players. Therefore, the boundary-following algorithm [2] is applied and the boundary of the field region is obtained. The field mask is obtained by filling the interior of this boundary. Figure 3 shows extracted field mask which is used to make player mask. Because all the players are on the field, we can find players in the field mask. After field extraction, the player mask $P(x,y)$

(figure 3) is made as follows.

$$P(x,y) = \begin{cases} 1 & : \quad \text{if } (x,y) \in \text{ field and } B(x,y) = 0 \\ 0 & : \quad \text{otherwise} \end{cases}$$

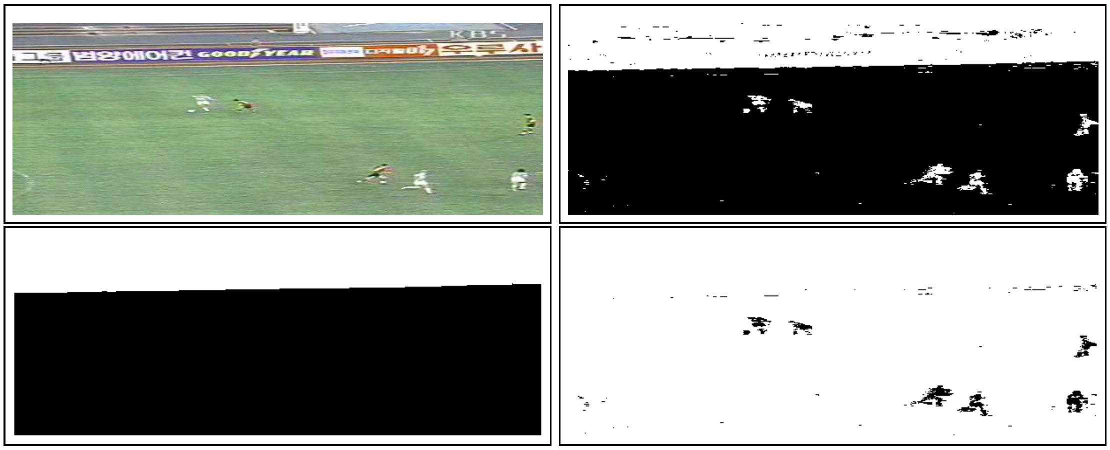


Figure 3: Upper left: original image. Upper right: binary image. Lower left: field mask. Lower right: player mask

3 Player and Ball Tracking

3.1 Player Tracking

For player tracking, we use template matching and Kalman filter [5]. The templates of players are extracted from player mask using connected component extraction. The typical templates are shown in figure 5. First, new players that do not significantly overlap with the bounding box of a player already tracked are found out. Then, the new players are inserted to tracking list. Locations of players at the next frame is predicted by Kalman filter and template matching at that location is performed. Finally, the player template is updated. Figure 4 is the whole procedure for tracking players. The figure 5 shows the process of Kalman filter based tracking.

The main problem of player tracking is occlusion and in this paper, we only consider occlusion between different teams. We use the *Histogram Backprojection* [7] for occlusion reasoning.

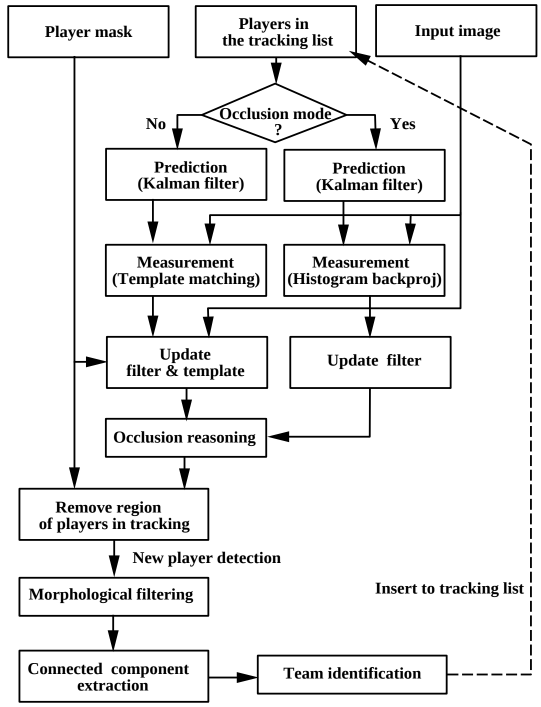


Figure 4: Procedure for tracking players

The ratio histogram R_i is computed from the histogram of the player template and image histogram:

$$R_i = \min\left[\frac{M_i}{I_i}, 1\right] \tag{1}$$

where, M_i is the histogram of player template, I_i is the histogram of input image and i denotes the index of each bin of the color histogram. Then this histogram R_i is backprojected onto the image and the backprojected image is convolved by a mask. The peak in the convolved image is the expected location of the player.

3.2 Team Identification

Kawashima [6] has analyzed the group behavior of soccer players using color histogram backprojection to isolate players on each teams. But since different team can have a similar histogram, we use vertical distribution of colors.

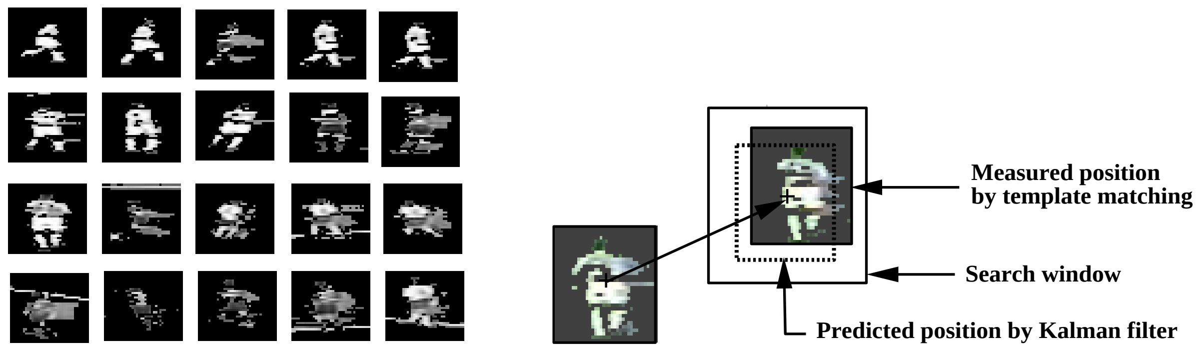


Figure 5: Left: examples of player template. Right: player tracking

1. Compute vertical distribution of R, G, B.

   Project player template horizontally and compute average values of R, G, B of each row. The length of the distribution is then normalized.

2. Compare this distribution with each team's model distribution.

   Model distribution is obtained when the first image is considered. The similarity measure is computed by convolution within a small range because the vertical location of a player in the template can vary slightly. Figure 6 shows the vertical distribution of color values. A similar distribution is obtained when players of the same team is given.

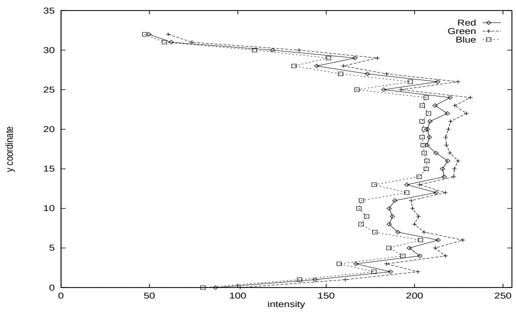
An experimental results are shown in figure 7 as different colors of bounding box.

3.3 Ball Tracking

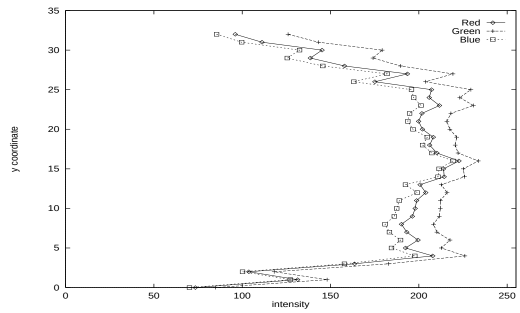
The method for ball tracking is similar to that of player tracking. But ball tracking is more difficult than player tracking since 1) automatic detection of the ball is very difficult because it is very small in the image, and 2) if a player has the ball, tracking is difficult because the ball is frequently occluded by the player. To solve these problems,

1. The position and bounding box of ball are manually initialized at the starting time.

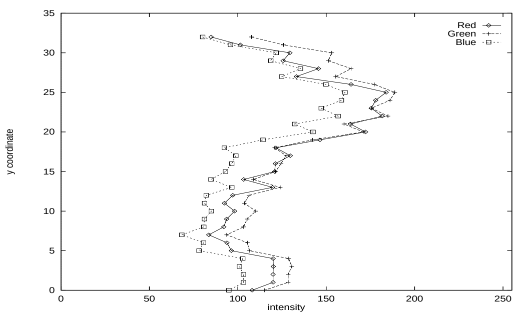
2. If a player is running near the ball, the player is marked ”has ball”. After that ball tracking has been stopped and the ball is searched around the player who has the ball. If the ball



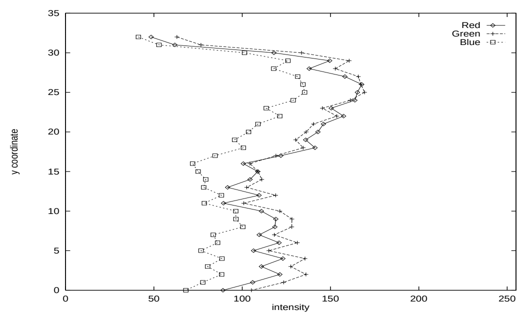
(a) player A



(b) player B



(c) player C



(d) player D

Figure 6: Vertical color distributions of templates of players. x-axis is the intensity and y-axis is the height of the plaer template. Player A and player B (C and D) have a similar distribution because they are the same team.

is found, we continue ball tracking. Search windows for various templates are shown in figure 7.

In figure 7, the ball position is indicated by an arrow and the image at the second row is the example of “when a player has the ball.”

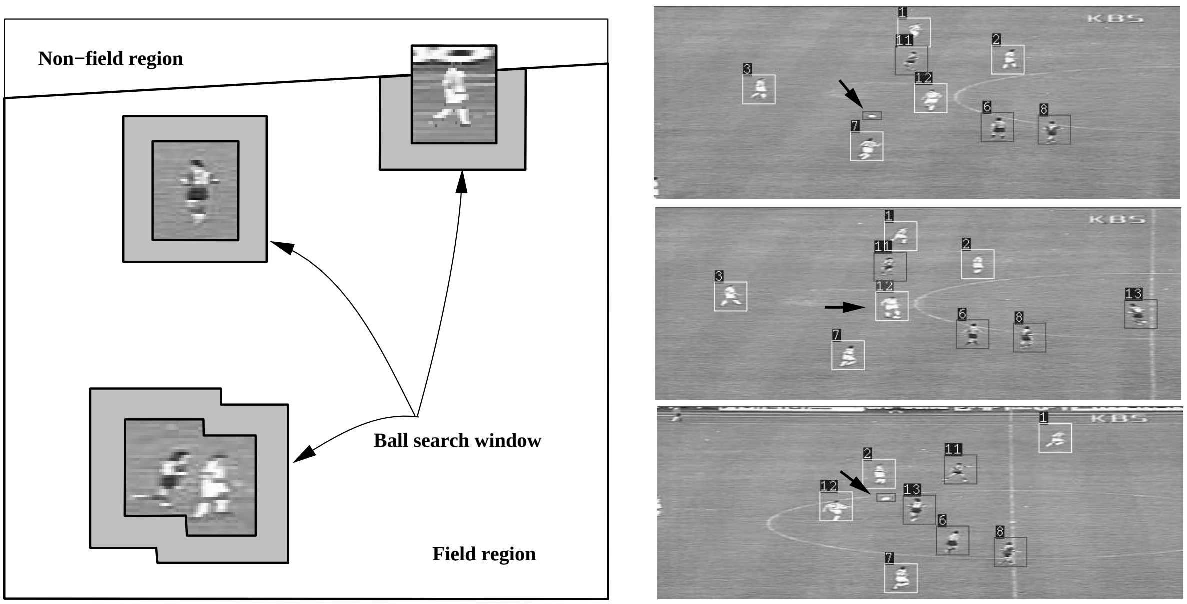


Figure 7: Ball tracking and players tracking. Left: Ball searching area around the bounding box of players is depicted. Right: The arrow positions the ball. In the second image the arrow denotes a player who has the ball. Each team is identified by boxes of different colors.

4 The Field Model and Player Position

In this section we describe the field model and how to get a transformation between the image and the model.

4.1 The Field Model

Figure 8 shows the field model. A player is located at certain position on the field. Since the camera is panning and zooming, the location will change from frame to frame. To find out the position of players on the field model, a transformation must be known between the image and the field model. If the center circle appears in the image, the transformation can be found. A simple four-point homographic planar transformation [1] can be used to map the image to the field model. From this transformation the player's position on the field model is obtained. First the vertical and horizontal line are found by Hough transform and next the ellipse is found [1]. Figure 9 shows the detected lines and the center circle. Then four points are found automatically and the image-to-model transformation is computed. The four points used to compute the transformation is shown in figure 8.In figure 9, the original image and the transformed image are shown. Right image of figure 8 shows the player position on the field model.

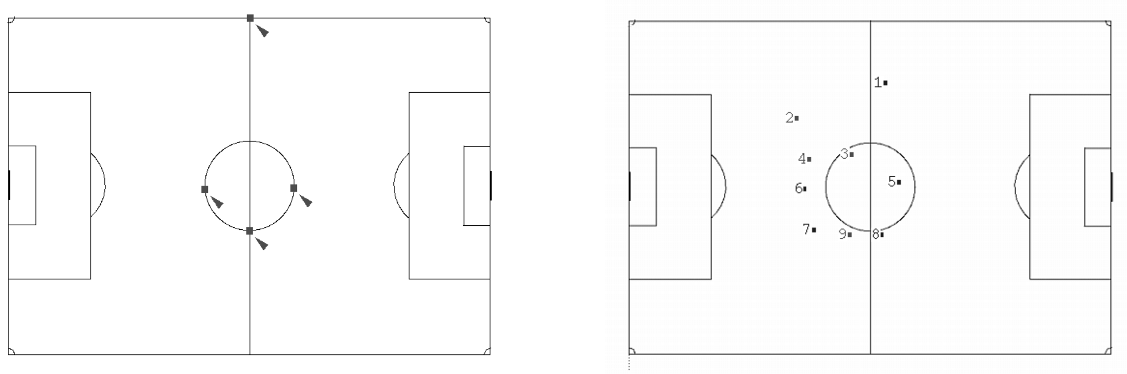


Figure 8: Left: Field model. The depicted four points are used to find the image-to-model transformation. Right: Player position on the field model. The image-to-model transformation allows to determine absolute positions of players. See figure 4.1

4.2 Soccer Image Mosaicking

In section 4.1, we have described how to compute the projective transformation between the image and the field model when the center circle is visible. But if the center circle is not visible, we cannot compute the image-to-model transformation. Figure 12 shows several frames of a soccer sequence. For the 1st frame and the 50th frame, the image-to-model transformation can be found. But for the 100th frame, the 120th frame, the 130th frame, and the 150th frame it is

[1] This transformation is a 3×3 matrix and has 8 essential parameters, since it is defined up to scale.

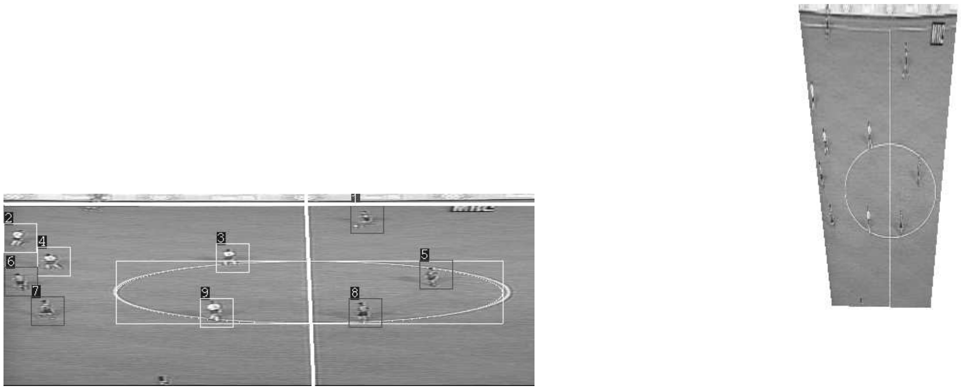


Figure 9: Left: Detected lines are the half line and the upper side line. Ellipse is denoted by its surrounding box. Right: Transformed image.

not easy to find the image-to-model transformation. We use image mosaicking method and by computing a transformation between reference frame and the field model, all frames are mapped to model.

To align two images we use the hierarchical direct registration technique [3]. First constructed are pyramids from the two processed images I_p^1 and I_p^2 of two input images, and then estimated are the motion parameters in a coarse-fine manner. Within each level the match measure $E(\mathbf{u})$ is

$$E(\mathbf{u}) = \sum_{\mathbf{X}} (I_p^1(\mathbf{x}, t) - I_p^2(\mathbf{x} - \mathbf{u}(\mathbf{x}), t-1))^2 \tag{2}$$

where $\mathbf{x} = (x, y)$: the image position, I_p : the processed image and $\mathbf{u}(\mathbf{x}) = (u(x,y), v(x,y))$: the image velocity. The procedure to compute I_p from input image I is given below. This measure is minimized with respect to 8 projective transformation parameters:

$$u(x,y) = \frac{p_1 x + p_2 y + p_5}{p_7 x + p_8 y + 1} \tag{3}$$

$$v(x,y) = \frac{p_3 x + p_4 y + p_6}{p_7 x + p_8 y + 1} \tag{4}$$

The function $E(\mathbf{u})$ is minimized via the Levenberg-Marquardt method. The processed image I_p is given through the following steps:

1. Make edge image(I_e) by sobel operation

2. Compute the histogram of the edge image and select threshold I_{th} at the 85% position of histogram

3. Threshold input image.

$$I_p(x,y) = \begin{cases} 0 & : \quad I_e(x,y) < I_{th} \\ I_e(x,y) & : \quad I_e(x,y) \geq I_{th} \end{cases} \tag{5}$$

4. Remove the region of players in the tracking list.

Figure 12 shows the mosaic image for 150 frames, too.

4.3 Player Position

In section 4.1, we found the transformation between image and model, and section 4.2 we constructed the mosaic image. Let's see figure 11. If we know transformation from image A to model(T_{MA}, by center circle) and image B to image A(T_{AB}, by mosaicking), the transformation from image B to model is given by $T = T_{MA}T_{AB}$. Thus, we get all player's position in the field model for all image sequences. Figure 10 shows the tracking result for 150 frames.

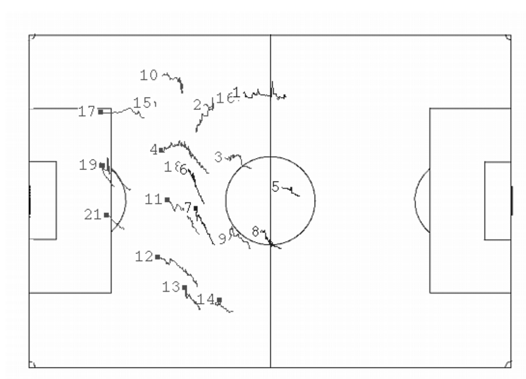


Figure 10: Tracking result for 150 frames

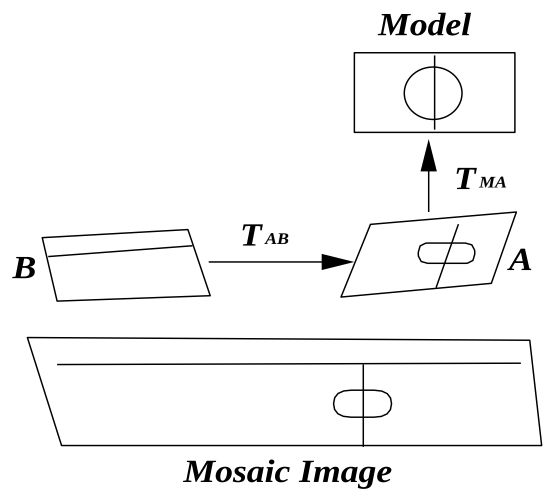


Figure 11: Mosaic image and image-to-model transformation

5 Conclusion

In this paper, we have shown how to track players and the ball, and have computed their absolute position on the real ground field. Addressed three main problems are ground field extraction, player and ball tracking and player positioning. Using real soccer image sequence we tested our method and the results are presented.[2] In future work, we are to develop algorithms which can deal with more complex problems as follows:

1. Occlusion reasoning of more complex case
2. Computing the absolute position of the ball during the play
3. More accurate computation of image-to-model transformation
4. View morphing or graphic synthesis at other viewpoints.
5. Exchanging Ad.'s
6. Automatic scaling of the size of player
7. Application to soccer game analysis

[2] One can see the experimental results in movie format. The files in SGI Movie file format can be obtained through Internet web site `http://cafe.postech.ac.kr`

References

[1] R. M. Haralick, L. G. Shapiro, *Computer and Robot Vision*, Addison Wesley, pp. 622-624, 1992.

[2] D. H. Ballard, C. M. Brown, *Computer Vision*, Prentice Hall, pp.143-146, 1982.

[3] M. Irani, D. Anandan, S. Hsu, *Mosaic Based Representation of Video Sequences and Their Applications*, ICCV' 95, pp. 605-611.

[4] S. S. Intille, A. F. Bobick, *Tracking using a local closed-world assumption: Tracking in the football domain*, MIT Media Lab Perceptual Computing Group TR No.296, MIT, August, 1994.

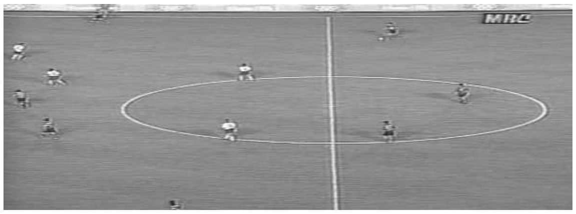
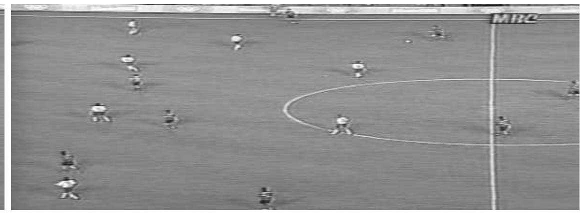
[5] A. Gelb, *Applied Optimal Estimation*, The MIT Press, pp. 102-119, 1979.

[6] T. Kawashima, K. Yoshino, Y. Aok, *Qualititative image analysis of group behavior*, In. Proc. Comp. Vis. and Pattern Rec., June 1994.

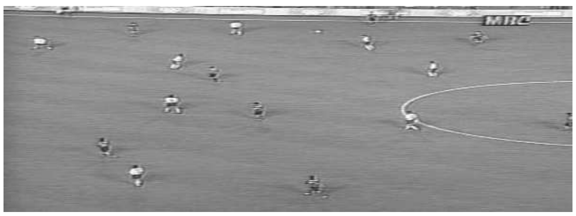
[7] M. J. Swain, D. H. Ballard, *Color indexing*, Int. Journal of Computer Vision, 7-1, pp. 11-32, 1991.

[8] T. Taki, J. Hasegawa and T. Fukumura, *Development of Motion Analysis System for Quantitative Evaluation of Teamwork in Soccer Games*, In Proc. of IEEE International Conference on Image Processing, Vol. III, pp. 815-818, 1996.

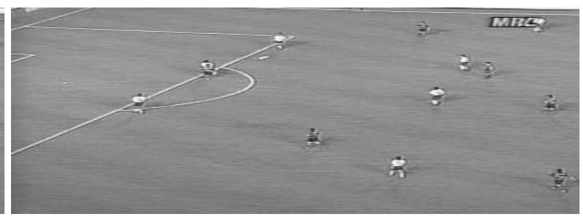
[9] D. Yow, B.L. Yeo, M. Yeung and B. Liu, *Analysis and Presentation of Soccer Highlight from Digital Video*, In Proc. of Second Asian Conference on Computer Vision, Vol. II, pp. 499-503, 1995.

1st frame 50th frame




100th frame 50th frame

130 frame 150th frame

Figure 12: Several frames of a soccer sequence and the mosaic image of them