Player Segmentation Evaluation for Trajectory Estimation in Soccer Games

Hanjoo Kim†, Siwok Nam ‡, and Jaihie Kim†
†Dept. of Electrical and Electronic Eng., Yonsei University,
Biometrics Engineering Research Center, Seoul, Korea
‡LG Electronics Institute of Technology, Seoul, Korea
hanjoo@yonsei.ac.kr

Abstract

In this paper, we evaluate the player segmentation for trajectory estimation in soccer games. In order to estimate the field trajectories of players in soccer games, we should accurately locate the foot positions of players in each soccer image and transform them into those in the soccer field. However, we cannot always segment the players completely, since players are often motion-blurred due to the fast motion of camera. We use k-means algorithm for accurate segmentation of the player's legs. Finally, we simulate the trajectory estimation for three different segmentation results: (i) when the legs of the players are accurately segmented; (ii) when the legs under the knees are missing; (iii) when only the torsos are segmented. Experimental results show that foot positions of the players should be located for accurate trajectory estimation.

Keywords: Segmentation, Trajectory, Homography

1 Introduction

Trajectory estimation of players on the ground from soccer image sequences has been actively studied since it gives a lot of information about the game[1][2][5]. In order to estimate field trajectories of players, we should accurately locate the foot positions of them. In the previous works, various player segmentation methods are used.

Choi *et al.* extracted players from soccer images by performing field extraction on the basis of color histogram[2]. Bebie *et al.* used multishape separation[1]. Kawashima *et al.* used the histogram backprojection[4]. Matsui *et al.* used color information[5]. Ohno *et al.* extracted regions of shirt and pants from images by using color information in the YIQ space[6]. However, in soccer image sequences, objects move fast and the camera continues panning and zooming. Hence we often miss the player's legs when segmenting the players due to the motion blur. In this paper, we use k-mean algorithm for accurate segmentation of the player's legs.

Once the foot positions of the players are obtained, we should find point-to-point correspondences between each soccer image and the soccer field, and this correspondence can be given by a homography. Homography, a 3×3 mapping matrix, can be determined from four corresponding point features provided that no three of them are collinear.

We evaluate the player segmentation results for trajectory estimation in soccer games. We conduct the trajectory estimation for three different cases of player segmentation and compare the experimental results. Experimental results with synthetic data show that the proposed segmentation method can provide more accurate trajectories than existing method.

2 Background

Homography is a linear transformation on homogeneous 3-vectors $\mathbf{x}_i$ and $\mathbf{x}'_i$ represented by a non-singular 3×3 matrix $\mathbf{H}$ as shown in Eq.(1). Given a set of $n(\geq 4)$ points $\mathbf{x}_i = (x_i, y_i, 1)^T$ in the projective plane $\mathbf{p}^2$ and a corresponding set of points $\mathbf{x}'_i = (x'_i, y'_i, 1)^T$ likewise in $\mathbf{p}^2$, we can compute the homography $\mathbf{H}$ that maps each $\mathbf{x}_i$ to $\mathbf{x}'_i$. Here,'$\simeq$' means equality up to a nonzero scale factor.

Homography can exist between a world plane and its image as follows. An image point is represented by a homogeneous 3-vector $\mathbf{x} = (x, y, 1)^T$, and world point by homogeneous 4-vector $\mathbf{X} = (X, Y, Z, 1)^T$. A scene point $\mathbf{X}$ is mapped to an image point $\mathbf{x}$ by perspective projection $\mathbf{P}$ as shown in Eq.(2). This map is represented by a 3×4 camera matrix $\mathbf{P}$, as $\mathbf{x} \simeq \mathbf{PX}$. If we choose the world coordinate system so that the first two axis define the plane, that is $Z = 0$, then $\mathbf{P}$ is reduced to a 3×3 homography between two projective planes $\mathbf{p}^2$[3].

$$\begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} \simeq \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \quad (1)$$

, *i.e.*, $\mathbf{x}'_i \simeq \mathbf{H}\mathbf{x}_i$, $i = 1, 2, \cdots, n(\geq 4)$.

$$\mathbf{x} = \mathbf{P}\mathbf{X} = \mathbf{K}\mathbf{R}\left[\mathbf{I}| - \tilde{\mathbf{C}}\right]\mathbf{X} \tag{2}$$

$$\text{where} \quad \mathbf{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The matrix $\mathbf{K}$ is defined up to a nonzero scaling, which is removed by setting $K_{33} = 1$ where the parameters are as follows: α_x, α_y is the focal length of the camera measured in pixel units along the horizontal and vertical directions respectively; (x_0, y_0) is the principal point of the camera, which is the intersection of the optical axis and the image plane, and is measured in pixels and can vary[3]; s is *skew factor*. The parameter contained in $\mathbf{K}$, α_x, α_y, s, x_0, y_0, are called the *internal* or *intrinsic* camera parameters of the camera. The rotation to align the camera coordinate system and the world coordinate system is given by $\mathbf{R}$.
Algebraically, if $\mathbf{x} = (x, y, 1)$, $\mathbf{x}' = (x', y', 1)$ are the images of a point $\mathbf{X} = (X, Y, Z, 1)$ before and after panning, tilting and zooming. Let Cartesian coordinate $\tilde{\mathbf{C}}$ represents the coordinate of the camera center in the world coordinate system. $\mathbf{I}$ is a 3×3 identity matrix. Let $\mathbf{P}$ and $\mathbf{P}'$ are the camera matrix before and after camera motion respectively. From Eq.(3), we can derive Eq.(5). From Eq.(5) and Eq.(6) we can derive Eq.(8). Finally, Eq.(8) can be written as Eq.(9). Consider two cameras with the same center as in Eq.(3). Note that there is a simple relation between them as in Eq.(9) since the cameras have a common center (see the derivation process from Eq.(3) to Eq.(8)).

$$\mathbf{P} = \mathbf{K}\mathbf{R}\left[\mathbf{I}| - \tilde{\mathbf{C}}\right], \ \mathbf{P}' = \mathbf{K}'\mathbf{R}'\left[\mathbf{I}| - \tilde{\mathbf{C}}\right] \tag{3}$$

$$\begin{aligned} \mathbf{P}' &= (\mathbf{K}'\mathbf{R}')(\mathbf{K}\mathbf{R})^{-1}\mathbf{K}\mathbf{R}\left[\mathbf{I}| - \tilde{\mathbf{C}}\right] && (4) \\ &= (\mathbf{K}'\mathbf{R}')(\mathbf{K}\mathbf{R})^{-1}\mathbf{P} && (5) \end{aligned}$$

$$\mathbf{x} = \mathbf{P}\mathbf{X} \tag{6}$$

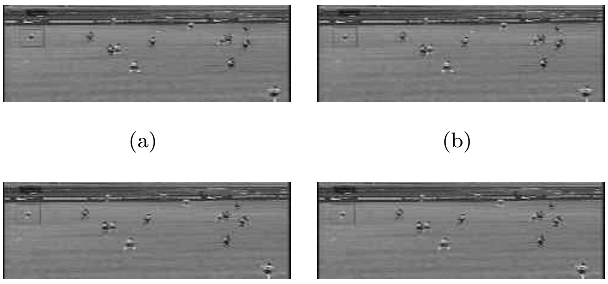
$$\begin{aligned} \mathbf{x}' = \mathbf{P}'\mathbf{X} &= (\mathbf{K}'\mathbf{R}')(\mathbf{K}\mathbf{R})^{-1}\mathbf{P}\mathbf{X} && (7) \\ &= (\mathbf{K}'\mathbf{R}')(\mathbf{K}\mathbf{R})^{-1}\mathbf{x} && (8) \end{aligned}$$

$$\mathbf{x}' = \mathbf{H}\mathbf{x}, \ \textit{where } \mathbf{H} = (\mathbf{K}'\mathbf{R}')(\mathbf{K}\mathbf{R})^{-1} \tag{9}$$

3 Soccer Player Segmentation

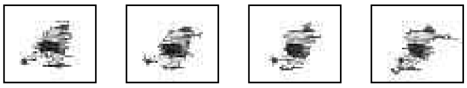
We extract a player by applying k-means algorithm in RGB space on template image. General soccer images have only 3 or 4 kinds of outstanding colors: color of shirt and pants region, color of lawn ground, etc. Applying clustering algorithm on the whole color soccer image is not feasible since the number of pixel of player region is too small. For the most part, segmenting player region fails due to small population of player region. Therefore, we apply clustering algorithm on template image. The k-means algorithm is implemented in the following steps:

1. Choose the number of cluster, k
2. For each of these k clusters, choose an initial cluster center $\{\bar{z}_1, \bar{z}_2, \ldots, \ldots, \bar{z}_k\}$
3. Distribute all pixels of a color image, *i.e.*, attach all pixels to the k clusters.
4. Calculate new cluster center with results of 3.
5. Check for convergence. If covergence has occured, stop. Otherwise iterate by going to step 3.



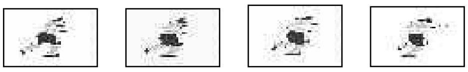
(a) (b) (c) (d)

Figure 1: Real soccer video sequences (a) The 1st frame, (b) The 2nd frame, (c) The 3rd frame, (d) The 4th frame



(a) (b) (c) (d)

Figure 2: The segmentation result of field extraction[2] (a) The 1st frame, (b) The 2nd frame, (c) The 3rd frame, (d) The 4th frame



(a) (b) (c) (d)

Figure 3: The segmentation result of k means algorithm (a) The 1st frame, (b) The 2nd frame, (c) The 3rd frame, (d) The 4th frame

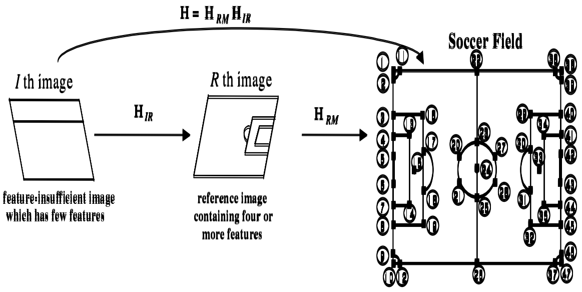


Figure 4: Overview of the trajectory estimation (Black rectangle indicate feature point)

The number of cluster, k, and choosing initial cluster centers, $\{\bar{z}_1, \bar{z}_2, \ldots, \ldots, \bar{z}_k\}$ are still challenging problems. We choose color of player's uniform and lawn ground as initial cluster centers. The templates in Fig.2 and Fig.3 are obtained from one player indicated by rectangle in Fig.1. As shown in Fig.2 and Fig.3, proposed method can segment the players more accurately.

4 Experimental Results

We conduct experiments with synthetic soccer data for quantitative evaluation. In situations there are many images of a scene, it is often required to compute the homography relating any given pair of images. A common method is to compute homographies only between temporally consecutive images in the sequence, and then use the concatenation of homographies to obtain the correspondence between temporally non-consecutive images[2]. We consider here a situation in which the camera rotates about its center but does not translate. Cameras used for broadcasts of soccer games are fixed in location but free to rotate and zoom. We simulate camera movements of soccer game in order to generate realistic synthetic soccer data.
We conduct experiments when the camera rotate(i.e., pan and tilt) about center with varying internal parameters(i.e., focal length). We generate synthetic data as follows.

1. Let $\mathbf{P}_i = \mathbf{K}_i\mathbf{R}_i\left[\mathbf{I}| -\tilde{\mathbf{C}}\right]$ is the camera matrix for the image i of the sequence. Given the sequence of N soccer images, we select the final image of sequence I_N and compute $\mathbf{P}_N$ using the feature points in the image N. And then we decompose $\mathbf{P}_N$ into $\mathbf{P}_N = \mathbf{K}_N\mathbf{R}_N\left[\mathbf{I}| -\tilde{\mathbf{C}}\right]$ using RQ decomposition[3] where $\mathbf{K}_N$ is internal parameter matrix, $\mathbf{R}_N$ is the rotation matrix, $\mathbf{I}$ is 3×3 identity matrix, and $\tilde{\mathbf{C}}$ is the coordinate of the camera center in the world coordinate system. Since finding $\mathbf{P}$ involves 11 independent parameters, at least 6 points are needed to determine $\mathbf{P}$. These points should be in general position[3], i.e.,the 6 points should not lie on the same plane. In this case, this implies that at least one reference point should not lie on the soccer field, and we use the top corners of the goal posts for this purpose.
2. Generate the camera matrix $\mathbf{P}_i(i = 1, 2, \cdots, N-1)$ by simulating the panning, tilting and zooming of the camera, i.e., by varying $\mathbf{K}$ and $\mathbf{R}$ for each image and, obtain $\mathbf{P}$ for each image of the sequence. According to $\mathbf{R}$ and $\mathbf{K}$, simulated image can be feature-insufficient or feature-sufficient.
3. Compute the consecutive interframe homography for feature-insufficient images using $\mathbf{H}_{i,i+1} = (\mathbf{K}_{i+1}\mathbf{R}_{i+1})(\mathbf{K}_i\mathbf{R}_i)^{-1}$. Gaussian noise with 0 mean σ standard deviation is added to the computed homographies since consecutive interframe homography is based on image measurement in the real soccer image sequences.
4. Generate the positions of objects in the soccer field. We simulate the trajectory estimation for three different segmentation results: (i) when the legs of the players are accurately segmented; (ii) when the legs under the knees are missing; (iii) when only the torsos are segmented. The coordinates of a foot point, a knee point and a pelvis point are given $(X, Y, 0)$, $(X, Y, 0.45)$, and $(X, Y, 0.9)$. $\mathbf{X}_i^j = (X_i^j, Y_i^j, 0), i = 1, 2, \cdots, N, j = 1, 2, \cdots, M$ means field position of object j in the i-th time instant of generating synthetic soccer data.
$\mathbf{X}^{obj}$=$\{(\mathbf{X}_1^1,\ \mathbf{X}_2^1,\ \cdots,\ \mathbf{X}_N^1),\ \cdots,\ (\mathbf{X}_1^M,\ \mathbf{X}_2^M,\ \cdots,\ \mathbf{X}_N^M)\}$
5. Obtain the image positions of objects in each image by applying camera matrix $\mathbf{P}_i(1 \leq i \leq N)$ to the generated field positions of objects in Step 1, and add Gaussian noise to the image positions. $\mathbf{x}_i^j = (x_i^j, y_i^j), i = 1, 2, \cdots, N, j = 1, 2, \cdots, M$ means image position of object j in the i-th time instant of generating synthetic soccer data.
$\mathbf{x}^{obj}$=$\{(\mathbf{x}_1^1,\ \mathbf{x}_2^1,\ \cdots,\ \mathbf{x}_N^1),\ \cdots,\ (\mathbf{x}_1^M,\ \mathbf{x}_2^M,\ \cdots,\ \mathbf{x}_N^M)\}$
6. Simulate the homography between the soccer field and the reference image $\mathbf{H}_{RM}$(see Fig.4).
7. Apply the given homographies $\mathbf{H}_{IR}$ and $\mathbf{H}_{RM}$ to the image positions of objects.

As illustrated in Fig.5, the result of foot position is most similar to the generated true trajectory.

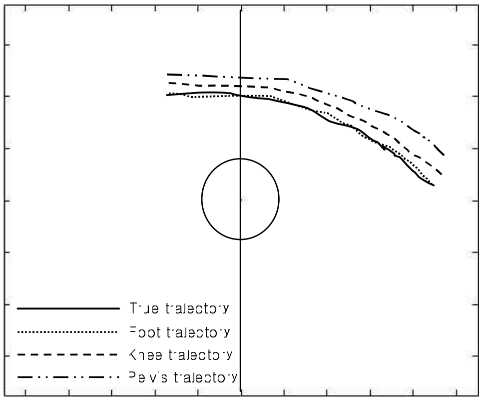


Figure 5: Experimental results

On the other hand, the trajectories of the knee points, and pelvis points are so different with the true trajectory. The RMS(root mean square) errors of foot, knee, and pelvis point for trajectory estimation are shown in Table.1. As shown in Table.1, accurate location of player's foot positions should be used for accurate trajectory estimation from uncalibrated soccer image sequences.

5 Conclusions

In this paper, we evaluate the effect of the segmentation results on the estimating trajectories of players in soccer games. In order to estimate the field trajectories of players in soccer games, we should locate the positions of players in each soccer image and transform them into those in the soccer field. However, we cannot always segment the players precisely, since players are often motion-blurred due to the fast motion of camera. We use k-means algorithm for accurate segmentation of the player's legs. Finally, we simulate the trajectory estimation for three different segmentation cases, i.e., when the legs of the players are accurately segmented; when the legs under the knees are missing; when only the torsos are segmented. According to the simulation results, trajectory estimation using foot points is more accurate than that using knee points and pelvis points. This paper show that accurate location of player's foot positions should be used for accurate trajectory estimation from uncalibrated soccer image sequences.

Table 1: RMS error between estimated and synthetic true trajectories(Unit:m)

Test No.	foot	knee	pelvis
1	0.45	2.95	5.23
2	0.33	2.23	5.44
3	0.34	2.88	5.17
4	0.63	2.34	5.52
5	0.53	2.38	5.36
6	0.39	2.79	5.12
7	0.36	2.87	5.31
8	0.34	2.41	5.34
9	0.38	2.13	5.15
10	0.48	2.45	5.33

Acknowledgements

"This work was supported (in part) by Korea Science and Engineering Foundation(KOSEF) through Biometrics Engineering Research Center(BERC) at Yonsei University."

References

[1] T.Bebie and H.Bieri, "A Video-Based 3D-Reconstruction of Soccer Games", EUROGRAPHICS 2000, Vol.19, No.3.

[2] S. Choi, Y. Seo, H. Kim and K. S. Hong, "Where are the ball and players ? Soccer game analysis with color-based tracking and image mosaik", International Conference on Image Analysis and Processing, Florence, Italy, 1997.

[3] R.Hartley and A.Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000.

[4] Toshio Kawashima, Kazuyoshi Yoshino and Yoshinao Aoki, "Qualitative Image Analysis of Group Behavior", International Conference on Computer Vision and Pattern Recognition, pp.690-693, 1994.

[5] K. Matsui, M. Iwase, M. Agata, T. Tanaka, and N. Ohnishi, "Soccer Image Sequence Computed by a Virtual Camera", IEEE Computer Society Conference on Computer Vision and Pattern Recognition 23 - 25 June, 1998.

[6] Y. Ohno, J. Miura and Y.Shirai, "Tracking Players and a Ball in Soccer Games", International Conference on Multisensor Fusion and Integration for Intelligent Systems, pp. 147-152, 1999.