MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan
14-4

Tracking Objects from Multiple Soccer Videos and Recognizing Events

Kyuhyoung Choi

Dept. of Media Technology
Sogang University
Seoul, Korea
kyu@sogang.ac.kr

Yongdeuk Seo

Dept. of Media Technology
Sogang University
Seoul, Korea
yndk@sogang.ac.kr

Abstract

This paper presents a novel way of recognizing events from a soccer match video sequence. After tracking players and the ball, events such as passing, kicking, having, scoring and struggling for the ball can be inferred for the subsequences. In our framework, the ball trajectory is represented as an image blob broken with cuts caused by occlusion or interference from a player. For those frame corresponding a cut, the ball is considered to become invisible. To address the ambiguity in determining whether a cut is caused by temporary occlusion or player's control, it is checked if the re-visible ball comes out in the tolerable range of the area where it is predicted by its dynamic model. If so, the cut is marked as temporary occlusion and virtually filled. For the resulted sub-blobs and cuts, events are extracted by maximum a posteriori estimation of the event makers. Experiments on sequences from two cameras shows satisfying results.

1. Introduction

Analysis of sports video sequences has been an interesting application in computer vision as the abundance of recent papers presents. Major topics of interests could be categorized as follows.

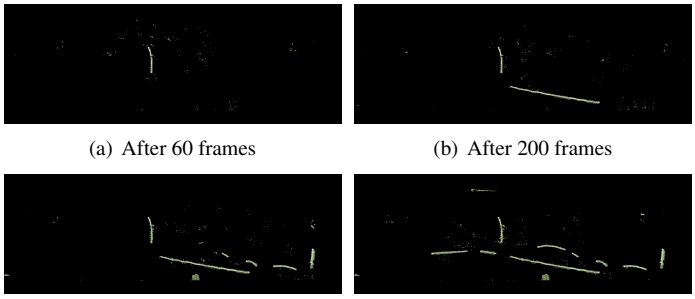
1. Tracking players and a ball in a sequence(s) captured by rotating/fixed cameras. Player tracking has been widely explored in [6, 18, 13, 3, 16, 9, 4, 7], and some papers such as [18, 17, 3, 15] have dealt with ball tracking problem as well.

2. 3D reconstruction of the scene and/or motion recovery of the cameras (self-calibration of a rotating and zooming camera) from the video sequence. Reid and Zisserman [14] and Kim *et. al.* [10] tried to estimate the 3D locus of the ball.

3. Augmented reality or re-synthesis in virtual space: [2, 12, 11]

4. Detection and recognition of events in video: [8, 1]

This paper focuses on analyzing a soccer video and extracting events for the subsequences. As a preprocessing, player and ball tracking is done and the results are used for the analysis [3]. Since the ball is a small object in the image, some parts of the ball trajectory are missing for frames where the ball is occluded by or near to players. Our work is both to supplement the incomplete trajectory and to give a description for the parts of a sequence. Section 2 introduces the algorithm of player and ball tracking and Section 3 describes how to extract events. Section 4 provides experimental results and finally Section 5 concludes this paper.



(a) After 60 frames (b) After 200 frames

(c) After 500 frames (d) After 650 frames

Figure 2: Accumulation images for the ball blobs.

2. Preprocessing

For sports video analysis, the preliminary step is tracking players (and ball). We used the player and ball tracking system presented in [13, 3].

Illustrated are the steps of image processing to automatically detect players and identify their classes[1]as in Figure 1. Particle filtering is used as the underlying tracker to give the trajectories of players and ball. Player tracking is processed first. Then for the ball tracking the less probable state subspace is taken out of consideration by exploiting the resulted player tracking and making an image of accumulated ball blobs as in Figure 2 [3].

3. Event Extraction

3.1 Event Moments

Before extraction, the state vector of a j th player, $\mathbf{i}(j)$ is defined as below.

$$\mathbf{i}(j) = (t(j), \mathbf{p}(j))^T$$
$$t \in \{team\_A, team\_B\}$$
$$\mathbf{p}(j) = (x(j), y(j))^T$$

where x and y is the horizontal and vertical coordinates in an image and t is one of two teams. F, a set of frame numbers, f where the ball(denoted as B) position could be estimated is defined as following.

$$F = \{f | \mathbf{p}_f(B) \neq \mathbf{0}\},$$

which means that $x(B)$ and $y(B)$ were set to be zero for the frames where the ball is not estimated. Due to the nature of

[1]e.g. player of team A, player of team B, goalie of team A, goalie of team B and referee

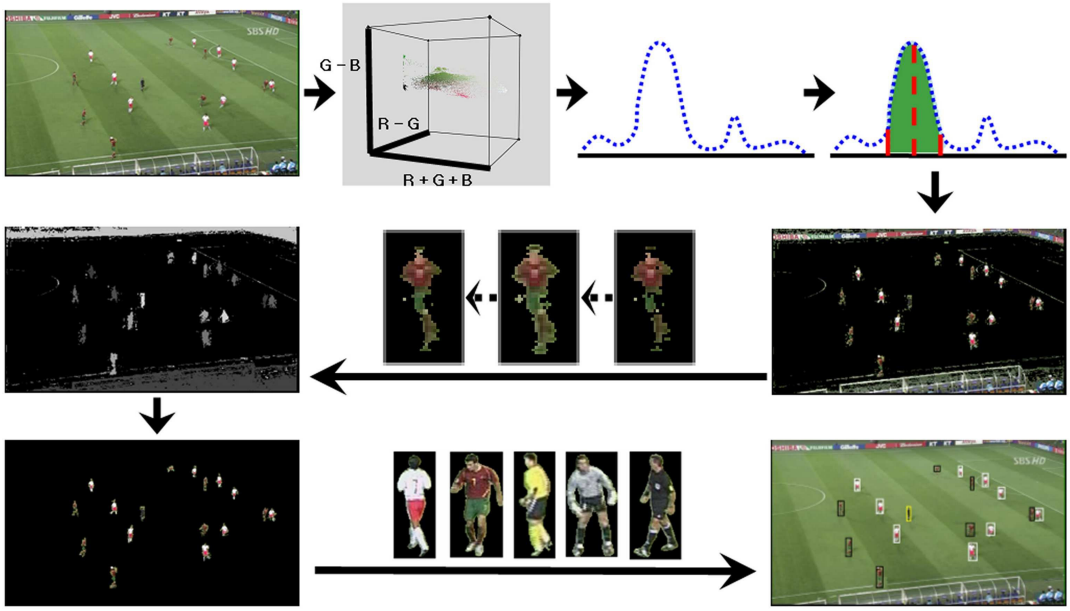


Figure 1: Image processing

ball games, the events can be rather detected by observing the ball than players. Though it depends one's personal viewpoint to take what level to define the events in, we took $PASS$, $KICK$, $HAVE$, $STRUGGLE$ and $SCORE$ for them. Again we classify them into two event sets as following.

$$\begin{aligned} I &= \{HAVE, STRUGGLE\} \\ V &= \{KICK, PASS, SCORE\} \end{aligned}$$

where I and V stand for *visible* and *invisible*, respectively. $PASS$ denotes the event that a player successfully passes the ball to his peer. $KICK$ happens when a player kicks the ball to an unspecified player or fails to make $PASS$. $HAVE$ is the event that the ball is fully in the control of a player, and hence *invisible*. $STRUGGLE$ happens when players of both team try to prevent the opponent making $HAVE$ and the ball is *invisible* at the same time. $SCORE$ is the event that the ball stays in a goal. Events happen at the moments of the ends of accumulated ball image blobs as shown in Figure 2. If the ball moves without intervention of players during a sequence, there will be one continuous trajectory blob in the image. So the ball trajectory is here represented as a series of sub-$TBLOB$s [2] with CUTs in between.[3] A CUT along the broken trajectory means the ball appears in the same blob of a player during the CUT, since he tries to control and change the ball motion. So a moment when an event occurs is at most among frames where $TBLOB$s begin and end, in other words, beginnings and ends of CUTs. However, some cuts are created when the ball just passes by a player, therefore, they can not be a moment of event. Hence the first step for event extraction is to find those cuts.

3.2 Cuts Due to Passing Through

To find a cut where there are no event change, we check the difference between ball motions before and after the cut. If the ball just passes by a player during the cut, the difference is supposed to be zero. However, due to the camera orientation and bounce of ball, a small amounts of differences are caused. The difference is measured as the mahalanobis distance between the ball position at the frame where a cut ends and the predicted position where the ball is supposed to reach with the velocity and acceleration at the beginning of the cut. The ordered set of frame numbers where the events really occurred, S is defined as follows.

$$S = \{f_{begin}(c), f_{end}(c) \,|D < Th, c \in C\}$$

where C is the set of all cuts in the sequence. Alternatively, S can be considered as a set of frame numbers of both ends of $TBLOB$s connected by this reasoning.

$$S = \{f_{begin}(z), f_{end}(z) \,|z \in Z\}$$

where Z is the set of the connected $TBLOB$s. For a CUT, c its length is the elapsed frames, $\Delta f(c)$, between the frames where the CUT begins and ends, $f_{begin}(c)$ and $f_{end}(c)$, respectively. If the ball is invisible during $\Delta f(c)$ due to a temporary occlusion, $\mathbf{p}_{f_{end}(c)+1}(B)$ is supposed to be $\hat{\mathbf{p}}_{f_{end}(c)+1}(B)$, the position to which it reaches with the velocity and acceleration at $f_{begin}(c) - 1$. Assuming Gaussian noise propagating through $\Delta f(c)$, we compute the mahalanobis distance D, between $\mathbf{p}_{f_{end}(c)+1}(B)$ and $\hat{\mathbf{p}}_{f_{end}(c)+1}(B)$.

$$\begin{aligned} D &= \mathbf{u}^T R^{-1}_{\Delta f(c)} \mathbf{u} \\ \mathbf{u} &= \left(\mathbf{v}_{f_{begin}(c)-1} \cdot \tilde{\mathbf{p}}(B), \mathbf{n}_{f_{begin}(c)-1} \cdot \tilde{\mathbf{p}}(B)\right)^T \\ \tilde{\mathbf{p}}(B) &= \mathbf{p}_{f_{end}(c)+1}(B) - \hat{\mathbf{p}}_{f_{end}(c)+1}(B) \\ R_{\Delta f(c)} &= \Delta f(c)\, LL^T \end{aligned}$$

where L is the covariance matrix of errors in pixel unit for a frame. If D is small, A $PCUT$ c is chosen to be due to passing through[4], that is, $f_{begin}(c)$ and $f_{end}(c)$ are removed from S.

[2] $TBLOB$ stands for trajectory blob.

[3] This can be seen as a series of CUTs and sub-$TBLOB$s in between also.

[4] $PCUT$ stands for a cut by passing through

3.3 Estimating Event Makers

Now it is certain that from $f_{begin}(z(i))$ until $f_{end}(z(i))$, one event of V happened and from $f_{end}(z(i))$ until $f_{begin}(z(i)+1)$[5], one event of I happened. Let us define $\mathbf{e}(z) = \left(\mathbf{i}_z^k, \mathbf{i}_z^r\right)$, a vector consisting of the state vectors of ball kicker and receiver $\mathbf{i}_z^k$ and $\mathbf{i}_z^r$ respectively, the event set $Q_{1:|Z|}$ is defined.

$$Q_{1:|Z|} = \{\mathbf{e}(z) | z \in Z\}$$

For a given sequence, the probability density of $Q_{1:|Z|}$,

$$\begin{aligned} p\left(Q_{1:|Z|}\right) &= p\left(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{|Z|}\right) && (1)\\ &= p\left(\mathbf{i}_1^r, \mathbf{i}_1^k, \mathbf{i}_2^r, \mathbf{i}_2^k, \ldots, \mathbf{i}_{|Z|}^r, \mathbf{i}_{|Z|}^k\right) && (2)\\ &= p\left(\mathbf{i}_1^r\right) p\left(\mathbf{i}_1^k | \mathbf{i}_1^r, \mathbf{i}_2^r\right) p\left(\mathbf{i}_2^r, \mathbf{i}_2^k, \ldots, \mathbf{i}_{|Z|}^r, \mathbf{i}_{|Z|}^k\right) && (3)\\ &= p\left(\mathbf{i}_1^r\right) p\left(\mathbf{i}_1^k | \mathbf{i}_1^r, \mathbf{i}_2^r\right) p\left(Q_{2:|Z|}\right) && (4)\\ &= \prod_{j=1}^{|Z|} p\left(\mathbf{i}_j^r\right) p\left(\mathbf{i}_j^k | \mathbf{i}_i^r, \mathbf{i}_{j+1}^r\right) && (5) \end{aligned}$$

Factorization from (2) to (3) are done using conditional probabilities. Because knowing the previous ball receiver $\mathbf{i}_1^r$ doesn't tell anything about the next receiver $\mathbf{i}_2^r$, $p(\mathbf{i}_2^r, \mathbf{i}_1^r)$ becomes $p(\mathbf{i}_2^r)$. It is also assumed that that state of $\mathbf{i}_1^r$ and $\mathbf{i}_1^k$ are independent from those of $\mathbf{i}_2^r$ and $\{\mathbf{e}(z(j)) | j \geq 3\}$. In (4), $p\left(Q_{1:|Z|}\right)$ is factorized into three factors the last of which is $p\left(Q_{2:|Z|}\right)$. Again $p\left(Q_{1:|Z|}\right)$ is represented as the product of probabilities corresponding to each CUT (5). Note that representation in (5) are a function of $\mathbf{i}_j^r$ and $\mathbf{i}_j^k$ if the state of $\mathbf{i}_{j+1}^r$ is given. Hence if we process reversely from an initially known $\mathbf{i}_{|Z|}^r$, maximum $p\left(Q_{1:|Z|}\right)$ is achieved, by maximizing the product of probabilities of each backward iteration and the event set $\hat{Q}$ is estimated as following.

$$\begin{aligned} \hat{Q} &= \max_{\mathbf{e}(i), i=1,2\ldots,|Z|} p(\mathbf{e}(1), \mathbf{e}(2), \ldots, \mathbf{e}(|Z|)) && (6)\\ &= \left\{ \max_{\mathbf{i}_j^r, \mathbf{i}_j^k} p(\mathbf{i}_i^r) p\left(\mathbf{i}_i^k | \mathbf{i}_i^r, \mathbf{i}_{i+1}^r\right) | j = 1, 2, \ldots |Z| \right\} && (7) \end{aligned}$$

The next thing is to define the probability densities $p\left(\mathbf{i}_j^r\right)$ and $p\left(\mathbf{i}_j^k | \mathbf{i}_j^r, \mathbf{i}_{j+1}^r\right)$. $p\left(\mathbf{i}_j^r\right)$ is a function of distance between the the ball and a candidate player at the beginning of the i th CUT. Since players too far from the ball is unlikely to stop the ball, an ellipse of validation gate is set with the center at the ball position. For a player $\mathbf{i}$ in a validation gate,

$$\begin{aligned} p(\mathbf{i}) &= p(t(\mathbf{i})) G(d(\mathbf{i}), \sigma)\\ p(t) &= \frac{1}{2}, t \in \{team\_A, team\_B\}\\ G(a, b) &= \frac{1}{\sqrt{2\pi} b} \exp\left(-\frac{a^2}{2b^2}\right) \end{aligned}$$

We define $p\left(t\left(\mathbf{i}_j^k\right) | t\left(\mathbf{i}_j^r\right), t\left(\mathbf{i}_{j+1}^r\right)\right)$ as follows.

$$p\left(\mathbf{i}_j^k | \mathbf{i}_j^r, \mathbf{i}_{j+1}^r\right) = p\left(t\left(\mathbf{i}_j^k\right) | t\left(\mathbf{i}_j^r\right), t\left(\mathbf{i}_{j+1}^r\right)\right) G\left(d\left(\mathbf{i}_j^k\right), \sigma\right) \quad (8)$$

On the other hand, when there is a goal detected (explained in Section 3.4) in a $TBLOB$ the computation of probability is slightly modified. Instead of using $p\left(\mathbf{i}_j^k | \mathbf{i}_j^r, \mathbf{i}_{j+1}^r\right)$ in (6), $p\left(\mathbf{i}_j^k | \mathbf{i}_j^r, \mathbf{k}_i\right)$ where $\mathbf{k}_i$ is the state of a goalie existing between $\mathbf{i}_j^k$ and $\mathbf{i}_{j+1}^r$(who is likely to be the same person, $\mathbf{k}_i$).

$$p\left(\mathbf{i}_j^k | \mathbf{i}_j^r, \mathbf{k}_j\right) = p\left(t\left(\mathbf{i}_j^k\right) | t\left(\mathbf{i}_j^r\right), t\left(\mathbf{k}_j\right)\right) G\left(d\left(\mathbf{i}_j^k\right), \sigma\right) \quad (9)$$

[5]Equivalently from $f_{begin}(c(i))$ until $f_{end}(c(i))$

3.4 Goal Detection

Since the most important highlight of ball games is the subsequence of goal, many studies in the field of semantic analysis and indexing of video have been done for goal detection. In our condition, the goal area is neither initially given nor estimated during tracking[6]or this post-processing even though we define the event $SCORE$ as in Section 3.1. Instead, we check the movements of players of each team after the ball passes by a goalkeeper as in Section 3.2. In the process of event maker estimation in Section 3.3, if a goalkeeper $\mathbf{k}$ is found to make a $PCUT$, c^* during a $TBLOB$, the ratio(denoted as r) of the sum of distances $h(t, f_{end}(c), f_{end}(c) + \Delta f), (t = \{team\_A, team\_B\})$ for which players of each team moves are computed.

$$\begin{aligned} r &= \frac{h(t^*, f_{end}(c), f_{end}(c) + \Delta f)}{h(t(\mathbf{k}), f_{end}(c), f_{end}(c) + \Delta f)}\\ h(t, f_0, f_0 + \Delta f) &= \sum_{i=1}^{N(t)} \sum_{f=f_0}^{f_0+\Delta f} \mathbf{n}_i^t(f)^T (s_i Y)^{-1} \mathbf{n}_i^t(f)\\ \mathbf{n}_i^t(f) &= \left(\mathbf{p}_i^t(f+1) - \mathbf{p}_i^t(f)\right) \end{aligned}$$

where Y is a metric matrix of image coordinate, s_i is a scaling factor depending on the y coordinate of i th player for considering the camera perspective projection and t^* is the opponent team of $t(\mathbf{k})$. A large value of r implies that the opponent players of the goalie has moved to each other or to the audience with joy of goal while players of the other team stand idle with disappointment.

3.5 Assigning Events to $TBLOB$s and CUTs

The last step, assigning events is relatively easy step since the event maker at each event moment has been figured out. The problem of assigning events $KICK, PASS, SCORE$ are actually finished through the previous steps. $KICK$ is assigned to frames of i th $BLOB$, $z(i)$ if $t\left(\mathbf{i}_j^k\right)$ and $t\left(\mathbf{i}_{j+1}^r\right)$ are the same, otherwise $PASS$. To differentiate between $HAVE$ and $STRUGGLE$ during i th CUT, if the $\mathbf{i}_j^r$ is estimated $\mathbf{i}^*$, $HAVE$ is assigned to the frames before another player comes close enough to $\mathbf{i}^*$. Then $STRUGGLE$ is assigned until the frame where the distance between them is large enough. The player whose position at the end of the CUT is assigned to be the next ball holder and $HAVE$ is assigned from this frame. This is done till the end of CUT.

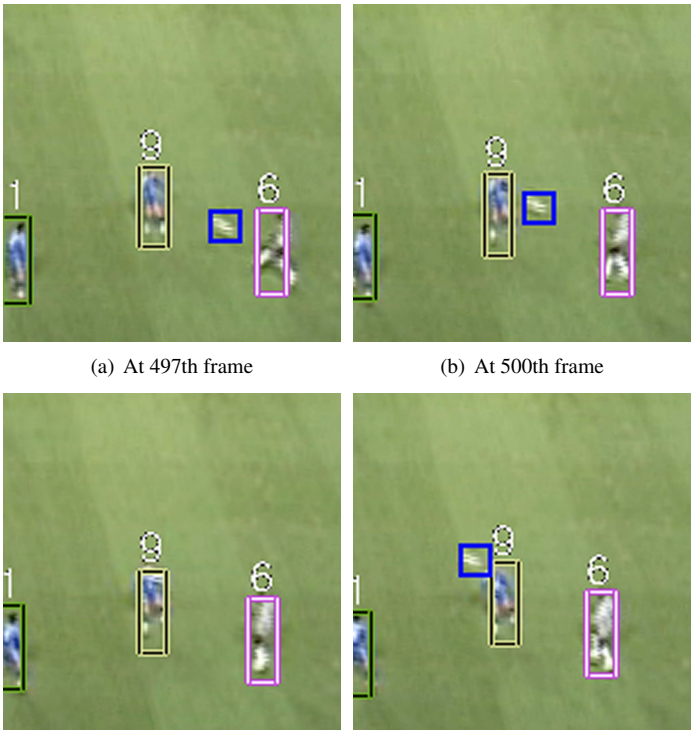
4. Experiments

In Figure 3 and Figure 4 some resulted subimages are shown for the cases of $PASS$ and $SCORE$ respectively. As it can be seen in the supplementary video clip, the frames otherwise labelled as $HAVE$ or $STRUGGLE$ are correctly classified as $PASS$ for subsequence of Figure 3. For the sequence of Figure 4, the scorer is correctly estimated even though his tracked position is not the closest to the ball with the conditional probability given in (8).

5. Conclusion

This paper presented a method for extracting events from soccer match video. Player and ball tracking results are exploited to analyze the sequence and classify the subsequences into some events. The classification is mostly based on the relative position

[6]For moving camera, this task becomes much more difficult



(a) At 497th frame (b) At 500th frame

(c) At 502th frame (d) At 504th frame

Figure 3: CUT by a temporary occlusion.

between the ball and players and the team ID of ball kicker and receiver.

Acknowledgments

This work was supported by the Korea Research Foundation Grant. (KRF-2004-003-D00356)

References

[1] M. Baillie and J. Jose. An audio-based sports video segmentation and event detection algorithm. In *IEEE Workshop on Detection and Recognition of Events in Video, Washington DC, USA*, 2004.

[2] T. Bebie and H.Bieri. A video-based 3D-reconstruction of soccer games. *Computer Graphics Forum*, 2000.

[3] K. Choi and Y. Seo. Probabilistic tracking of the soccer ball. In *Int. Workshop on Statistical Methods in Video Processing, in conjunction with ECCV 2004, Prague, Czech Republic*, 2004.

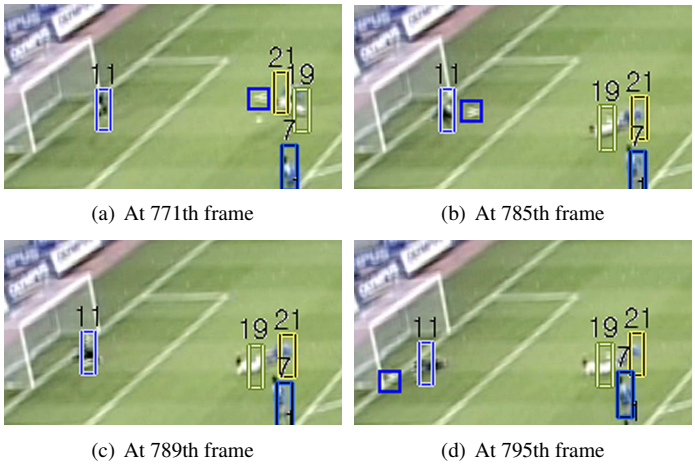
[4] P. Figueroa, N. Leite, R.M.L. Barros, I. Cohen, and G. Medioni. Tracking soccer players using the graph representation. In *ICPR04*, pages IV: 787–790, 2004.

[5] N. Inamoto and H. Saito. Immersive observation of virtualized soccer match at real stadium model. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2003.

[6] S. Intille and A. Bobick. Closed-world tracking. In *Proc. Int. Conf. on Computer Vision*, 1995.

[7] S. Iwase and H. Saito. Parallel tracking of all soccer players by integrating detected positions in multiple view images. In *ICPR04*, pages IV: 751–754, 2004.

[8] E. Jaser, J. Kittler, and W. Christmas. Hierarchical decision making scheme for sports video categorisation with temporal post-processing. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.



(a) At 771th frame (b) At 785th frame

(c) At 789th frame (d) At 795th frame

Figure 4: Subsequence of a event $SCORE$.

[9] J. Kang, I. Cohen, and G. Medioni. Soccer player tracking across uncalibrated camera streams. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, 2003.

[10] T. Kim, Y. Seo, and K.S. Hong. Physics-based 3d position analysis of a soccer ball from monocular image sequences. In *Proc. Int. Conf. on Computer Vision*, pages 721–726, 1998.

[11] T. Koyama, I. Kitahara, and Y. Ohta. Live mixed-reality 3d video in soccer stadium. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2003.

[12] K. Matsui, M. Iwase, M. Agata, T.T. Tanaka, and N. Ohnishi. Soccer image sequence computed by a virtual camera. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 860–865, 1998.

[13] H.W. OK, Y. Seo, and K.S. Hong. Multiple soccer players tracking by condensation with occlusion alarm probability. In *Int. Workshop on Statistically Motivated Vision Processing, in conjunction with ECCV 2002, Copenhagen, Denmark*, 2002.

[14] I. Reid and A. Zisserman. Goal-directed video metrology. In *Proc. European Conf. on Computer Vision*, 1996.

[15] X.F. Tong, H.Q. Lu, and Q.S. Liu. An effective and fast soccer ball detection and tracking method. In *ICPR04*, pages IV: 795–798, 2004.

[16] O. Utsumi, K. Miura, I. IDE, S. Sakai, and H. Tanaka. An object detection method for describing soccer games from video. In *IEEE International Conference on Multimedia and Expo (ICME)*, 2002.

[17] 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. In *Proc. International Conference on Pattern Recognition*, 2002.

[18] D. Yow, B. Yeo, M. Yeung, and B. Liu. Analysis and presentation of soccer highlights from digital video. In *Proc. Asian Conf. on Computer Vision*, 1995.