Event Detection and Recognition Using HMM with Whistle Sounds

Hiroki ITOH
Graduate School of System Informatics
Kobe University
Kobe, Japan
Email: itoh@me.cs.scitec.kobe-u.ac.jp

Tetsuya TAKIGUCHI, Yasuo ARIKI
Organization of Advanced Science and Technology
Kobe University
Kobe, Japan
Email: {takigu,ariki}@kobe-u.ac.jp

***Abstract*—In this paper, we propose a new method to detect and recognize events robustly in a soccer game. Based on the players density and speed, the events are detected and recognized using Hidden Markov Model (HMM). However, it is difficult to detect "free kick" and "throw in" because these events occur anytime and anywhere. In a soccer game, some event occurs when the referee blows a whistle or a ball is out of field. Therefore, we improve the detection accuracy of the events such as "free kick" and "throw in" by using these information when they occur. Also, event recognition is performed by an integration method of the results obtained using two types of HMMs : one is for players and the other is for a ball.**

***Keywords*-Hidden Markov Model (HMM); event detection; whistle sounds; player tracking; soccer game**

I. Introduction

Recently, digital photographing by amateurs has been widely spread so that any one can shoot sports games easily. Therefore, the need for video editing is getting strong, and automatic video production systems are attracting attention, which support video editing based on individual preference. These systems are composed of image recognition techniques to track the players and ball [1], [2], event detection, and digital camera work [4]. Event detection is the key issue for digital camera work as well as for retrieving the event and summarizing the whole soccer game.

Previously we have developed the event detection and recognition system in a soccer game based on rule-base voting [5]. This method assumed that the events in a soccer game were uniquely defined by the position of players and a ball. Specifically, a rule set was prepared in advance to associate the position information of players and a ball to an event. An event was detected by calculating the matching rate of the tracking result of players and a ball with rule set at every frame. However, there was a problem that the detection accuracy of the events such as the free kick and throw-in was low since these events occurred anytime and anywhere on the field and the occurring duration was also short. Moreover, many parameters of the rule sets had to be tuned manually.

On the other hand, Motoi [6] detected the sports event using Hidden Markov Model. This method employed General Hidden Markov Model, which could express the time dependency in time series data. The players position information was used as the feature. It was not necessary to set the parameter of the rule sets as conventional methods with rule-base voting, and could be applied to other sport video images. However, it could not improve the detection accuracy of the events such as free kick or throw-in. Therefore, in this paper, we propose a new method to detect and recognize these events robustly in a soccer game based on the stochastic model, HMM.

This paper is organized as follows. In section 2, the flow of the proposed method is described. Section 3 describes players tracking. In section 4, the details of event detection and recognition are described. In section 5, the performance of the proposed method is evaluated for the actual and simulation videos. Section 6 is for paper summarization and discuss about the future work.

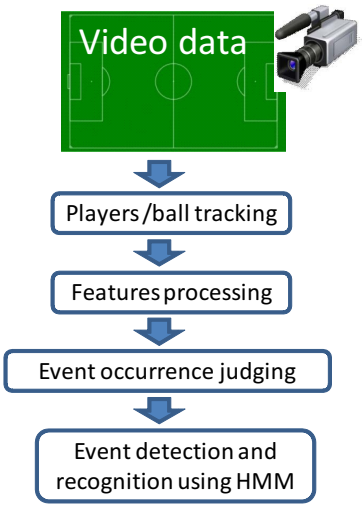

Figure 1. Flow of event detection and recognition

II. Flow of the Proposed Method

Figure 1 shows a flow of event detection and recognition. Firstly, the players and a ball are tracked in a video data. The details of players tracking system are described in the next section. The ball tracking system [7] uses a switching

978-1-4799-3211-5/13 $31.00 © 2013 IEEE
DOI 10.1109/SITIS.2013.14

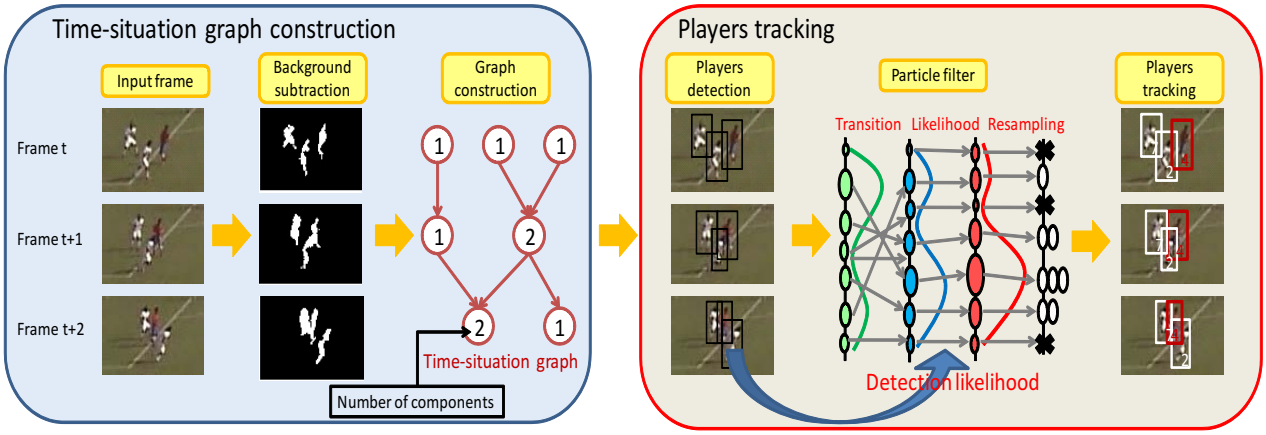

Figure 2. Flow of the tracking system

method between global search and local search adaptively. Secondly, the players density is computed as the feature in the small squared area produced by dividing the soccer field, and averaged speed of the players is also computed. Thirdly, the event occurrence is judged by whistle sounds and the ball position(when the ball is out of the field). Finally, each event is detected and recognized by HMM.

III. Players Tracking

Many tracking methods have been proposed previously such as mean-shift, Kalman filter [3], covariance tracker and particle filter [9]. Especially, particle filter was often employed in tracking players in a soccer image sequence. However, when it loses the players once, it is difficult to discover them again since the position information of two or more players can not be used between the frames in an image sequence.

In this paper, we represent the position information of two or more players as the time-situation graph beforehand. Then, by running particle filter guided by this graph, we can greatly reduce the incorrect detection of players and track players robustly even when occlusion occurs.

A. Flow of Players Tracking

Figure 2 shows a flow of our method which is composed of the time-situation graph construction process and the players tracking process. In the time-situation graph construction process, each player area is first extracted from an input frame by background subtraction. Then, the time-situation graph is constructed by storing into the node the information extracted in the player area. The time-situation graph is a graph whose node includes the number of players (henceforth, it is called the number of components) who exist in a player area extracted by background subtraction at every frame.

In the players tracking process, all players are first detected according to the node information in the time-situation graph. Then, each player is tracked by 3D particle filter in the detection area guided by the time-situation graph.

B. Time-Situation Graph

We represent the position information of two or more players as the time-situation graph beforehand using the method proposed by Figueroa [10]. In this method, the players, who are occluded each other, can be correctly detected since the number of players who exist in a player area is obtained in a time-situation graph shown in Figure 2(after background subtraction). Then, by running particle filter in the form guided by this graph, it is expected to reduce the incorrect detection of players greatly and to track players robustly even when occlusion occurs.

Table I
TIME-SITUATION GRAPH INFORMATION

Node information	Edge information
Label	Label
Area	Distance between nodes
Size(width and height)	
Position	
Number of components	

C. Time-Situation Graph Information

The time-situation graph is constructed based on a set of player areas extracted at every frame by background subtraction as shown in Figure 2(in graph construction process). Table I summarizes the node information and edge information. The node information is composed of the label to identify each node, player area(the number of pixels), player area size(width and height), the center point coordinates of the player area and the number of components. The edge information is composed of the label to identify each edge and the distance between nodes.

The color information was included in the edge information in the method proposed by Figueroa [10] because

their method tracked players only using graph. However, the time-situation graph which we use in this paper detects the players based on the number of components. Therefore, the necessary operation in this graph is only to make the number of components change. From this point, the color information is not included in the edge information in our method.

1) Construction of the Time-Situation Graph: Let G be a time-situation graph which is a directed graph. $n_i(t)$ is a node labeled i (identification number) at frame t. $d_{i,j}$ is the distance between nodes $n_i(t)$ and $n_j(t+1)$. $d_{max(i,j)}$ represents the estimated max distance to which a player can move within 2 frames, and this is estimated according to 3D world coordinate position. The distance between nodes is defined by the Euclid distance between the center point coordinates of the player area at frame t and the center point coordinates of the player area at frame $t+1$.

In Figure 3, the distances are shown between node $n_1(t)$ and nodes at frame $t+1$. The yellow point at frame $t+1$ locates in the same position of node $n_1(t)$ at frame t. The red line segments which stretch from the yellow point to the center point coordinates of each nodes at frame $t+1$(the blue points) are the distances between nodes. $e_{i,j}$ is the edge between nodes $n_i(t)$ and $n_j(t+1)$.The algorithm to construct the time-situation graph is defined according to the following steps.

1. Create a node $n_i(t)$ for the node labeled i at the first frame(t = 1), and insert this node into a time-situation graph G.
2. Create a node $n_j(t+1)$ for the node labeled j at frame $t+1$, and insert this node into the time-situation graph G.
3. Calculate the distance $d_{i,j}$ between nodes $n_i(t)$ and $n_j(t+1)$.
4. Create an edge $e_{i,j}$ satisfying the condition $d_{i,j} < d_{max(i,j)}$.
5. Group the nodes, and determine the number of components based on the player area.
6. Repeat steps 2-5 for the whole image sequence.

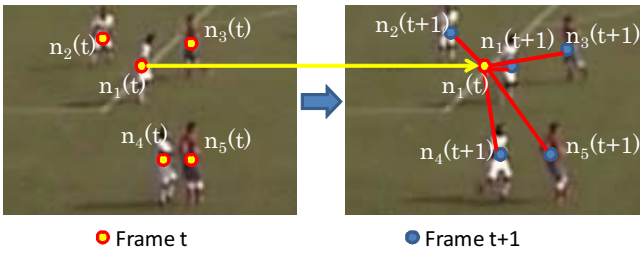

Figure 3. Distance between nodes

2) Determination of the Number of Components:

Grouping Nodes: This section describes the method of grouping the nodes by using the edge between them as shown in Figure 4. When node $n_{w_4}(t+1)$, to which edges e_{v_3,w_4} and e_{v_4,w_4} are linked, exists at frame $t+1$, two nodes $n_{v_3}(t)$ and $n_{v_4}(t)$ at frame t belong to the same group as $n_{w_4}(t+1)$. In other words, all the nodes linked by edges belong to the same group.

A new group number is defined for every new group detected on the time-situation graph. For example in Figure 4, starting from node $n_{v_1}(t)$, Group1 is detected, and in the same way starting from node $n_{v_3}(t)$, Group2 is detected.

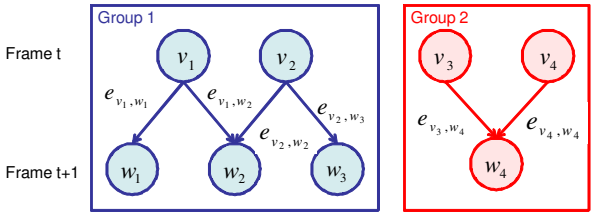

Figure 4. Grouping nodes

Determination of the Number of Components: In the proposed method, the determination of the number of components plays an important role because the players are detected based on the number of components. The top in Figure 5 shows the example where three players are occluded each other and the bottom in Figure 5 shows the flow of the determination of the number of components. The steps are summarized in the following algorithm.

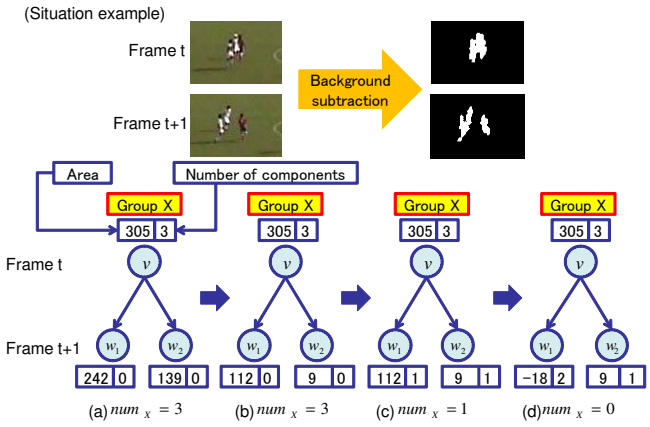

Figure 5. Flow of the determination of the number of components

Algorithm

[Step1] num_X (the number of components for GroupX) is first defined as the total number of components included in all nodes in GroupX at frame t. The number of components for each node at feame $t+1$ is still 0.

$$num_X = \sum_{v_i \in X} (num_{v_i})$$

Each player area is measured after background subtraction. In Figure 5(a), num_X is set to 3, and sizes of the the player areas are 305, 242 and 139.

[Step2]The ideal player area A_p for one player is estimated by the position information of each node $n_{w_i}(t+1)$ in GroupX at feame $t+1$. Each node $n_{w_i}(t+1)$ is updated by using this ideal player area A_p as follows.

$$A_{w_i} \leftarrow A_{w_i} - A_p$$

In Figure 5(b), The estimated ideal player area A_p is 130, and player areas at frame $t+1$ are updated from 242 to 112 (242-130) and 139 to 9 (139 -130) respectively.

[Step3]The number of components num_X is decreased by 1, and the number of components for each node $n_{w_i}(t+1)$ at frame $t+1$ is initialized by 1. This is repeated for every node at frame $t+1$.

$$num_X \leftarrow num_X - 1$$

$$num_{w_i} = 1$$

In Figure 5(c), the number of components num_X for GroupX is decreased by 2, and the number of components for each node at frame $t+1$ is set to 1.

[Step4]The following process is repeated to the node $n_{w_i}(t+1)$ whose area A_{w_i} is largest among the players at frame $t+1$ until the number of components num_X for GroupX is 0.

$$A_{w_i} \leftarrow A_{w_i} - A_p$$

$$num_X \leftarrow num_X - 1$$

$$num_{w_i} \leftarrow num_{w_i} + 1$$

In Figure 5(d), the number of components num_X is 0, and each number of components num_{w_i} at frame $t+1$ is finally determined.

D. Tracking Method

1) Players Detection Based on Node Information: Since the "tracking-by-detection" method [8] which combines players tracking with players detection is employed in this paper, the accuracy of players detection has a great effect on the tracking accuracy. The method of players detection differs according to the number of components contained in the node of the time-situation graph.

In the case where the number of components is 1 or 2, the players are detected based on using player area information of the node. In the case where the number of components is 3 or more, the players are detected by using player area information of the node and SVM. First, as shown in Figure 6(a) and (b), the vertically locating 2 players are detected based on the y coordinate between mini(y_{min}) and max(y_{max}), and the detection window size is estimated by 3D world coordinates position corresponding to the center coordinates of the player area, included in the node. Then, as shown in Figure 6(c), the other players are detected by using SVM [12] which is widely used in various applications and is known for recognition performance being high.

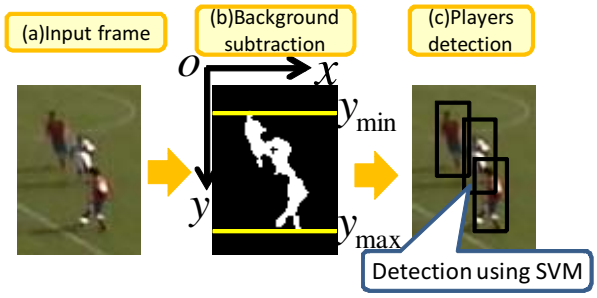

Figure 6. Players detection

2) Players Tracking by 3D Particle Filter Using Time-Situation Graph: We employed a 3D particle filter for each player tracking. The state $\vec{x}_p(t)$ at time t is defined as follows:

$$\vec{x}_p(t) = [p_x, p_y, v_x, v_y, a_x, a_y]^T \quad (1)$$

Here, p, v and a are the positions, velocities and accelerations at time t. Then, the state transition of the player is modeled based on linear motion with uniform acceleration.

The likelihood ω of the player is computed at each particle by the weighted sum as shown in Eq (2) using three kinds of likelihoods (the likelihood p_D based on the detection result by the node information and SVM, the likelihood p_H based on the histogram and the likelihood p_C based on the cross-correlation). Here, α is defined as the parameter showing the degree of occlusion, and computed based on the node information(area) obtained when the player was detected. β is defined as the value obtained by converting the distance between the target player and the nearest player into the probability of the normal distribution.

$$\omega = \alpha \cdot p_D + \beta \cdot p_H + (1-\beta) \cdot p_C \quad (2)$$

IV. Event Detection and Recognition

Events in soccer game can be defined using the position of the players and ball. For example, when "corner kick" occurs, the ball exists in the corner arc and the players of both teams crowd around the goalmouth. Therefore, the accuracy of event detection depends on the accuracy of players and ball tracking. Moreover, it becomes possible to detect the event more accurately by recognizing the referee whistle sounds or by judging whether a ball is out of field when an event occurs.

A. Features Processing

In our method, events are detected and recognized by HMM. Table II summarizes the players feature and the ball feature used in HMM. The players feature is composed of

Figure 7. Flow of the result integration method using two types of HMM

the number of players in 16 grids produced by dividing the soccer field as shown in Figure 8 and the averaged speed of all players. The ball feature is composed of a ball position, speed and grid number where a ball exists.

Table II
DETAILS OF FEATURES EMPLOYED FOR HMM

Players feature (18dim)	Ball feature (5dim)
Number of players in each grid (16dim) Averaged speed (2dim)	Position (2dim) Speed (2dim) Grid number (1dim)

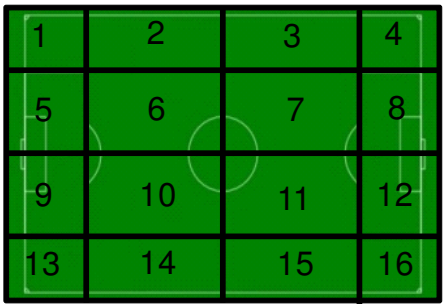

Figure 8. A soccer field divided into 16 grids

B. Event Occurrence Judgment

In a soccer game, some events occur when the referee blows a whistle or a ball is out of field. We use these information to estimate the occurrence time in event detection. Hence, event occurrence is recognized by the judgment whether whistle sounds exist or the judgment whether a ball is out of field. Whistle sounds can be recognized by HMM, and whether a ball is out of field is recognized by using the tracking result of a ball.

C. Event Detection and Recognition Using HMM

We employ the result integration method using two types of HMM (HMM based on the players information and HMM based on the ball information) as shown in Figure 7. A final likelihood is computed as shown in Eq (3).

$$L_{b+p} = \alpha L_b + (1-\alpha)L_p , \quad 0 \leq \alpha \leq 1 \qquad (3)$$

where L_{b+p} is a likelihood after integration, L_b and L_p are likelihoods of ball and players features respectively. α is the combination weight.

V. EXPERIMENTAL EVALUATION

We performed two experiments, the players tracking experiment and the event detection and recognition experiment.

A. Evaluation of Players Tracking

We selected a soccer game that was played during the 38th National High School Soccer Championship (Kyoto area final) in Japan. The video was taken with a fixed camera. The size of the image was 1280*720 pixels with 24-bit color.

In players tracking, we compared our method using the time-situation graph with the conventional method [11] without the time-situation graph for 10 videos (the average number of frames : 350) clipped from the soccer video. The tracking space is the only left half of the field. Therefore, we selected the sample videos in which the players almost played in the left half of the field since the players were not able to be tracked when they went out into the right half of the field.

The conventional method for the players detection employed SVM. The number of particles were 150, and the frame rate was 30 fps. The results are shown in Figure 9. In Figure 9, "Tracking accuracy" is the ratio of the number of correctly tracked frames to the number of the total frames. Tracking accuracy by our method is almost higher than that of the conventional method without the time-situation graph.

As a result, the tracking accuracy of our method is improved by 7.15 points on the average as shown in Table III. As can be seen from this result, since the accuracy of tracking players is higher by using time-situation graph when

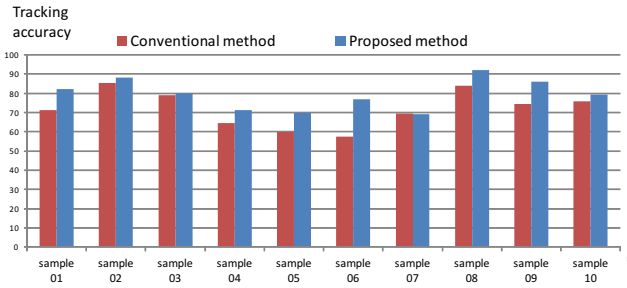

Figure 9. Comparison between the two methods

occlusion occurs, players tracking becomes more robust than the conventional method. Tracking failure occurs when the players of the same team cross each other so that their particles are reversed or overlap to 1 player. For the former, it can be thought that the likelihood of the particle filter does not decrease due to the same uniform color of the same team even if tracking is reversed. For the latter, it can be thought that the number of components in the time-situation graph becomes wrong. The proposed method is inferior to the conventional method in the point that the players with no movement for long frames may not be detected by background subtraction. However, our method can more correctly detect the players than the conventional method when their occlusion occurs.

Table III
AVERAGED TRACKING ACCURACY

Method	Rate (%)
Conventional method	72.15
Our method	79.50

Figure 10 shows the tracking results clipped at fixed intervals. It turns out that the occluded players are tracked correctly.

B. Evaluation of Event Detection and Recognition

1) Condition of HMMs: Table IV shows the condition of HMMs employed in the experiment.

Table IV
THE CONDITION OF HMMS

Type of HMM	Number of states	Number of mixtures
Whistle sounds	3	2
Players	3	6
Ball	3	6

2) Evaluation of Whistle Sounds Recognition: We used actual audio sequence with 13 minutes for the experiment of whistle sounds recognition. In this audio sequence, there were 18 whistle sounds in total. MFCC(12dim) was used as the feature of HMM .

Table V shows the accuracy of the whistle sounds recognition. Although all whistle sounds in an audio sequence can

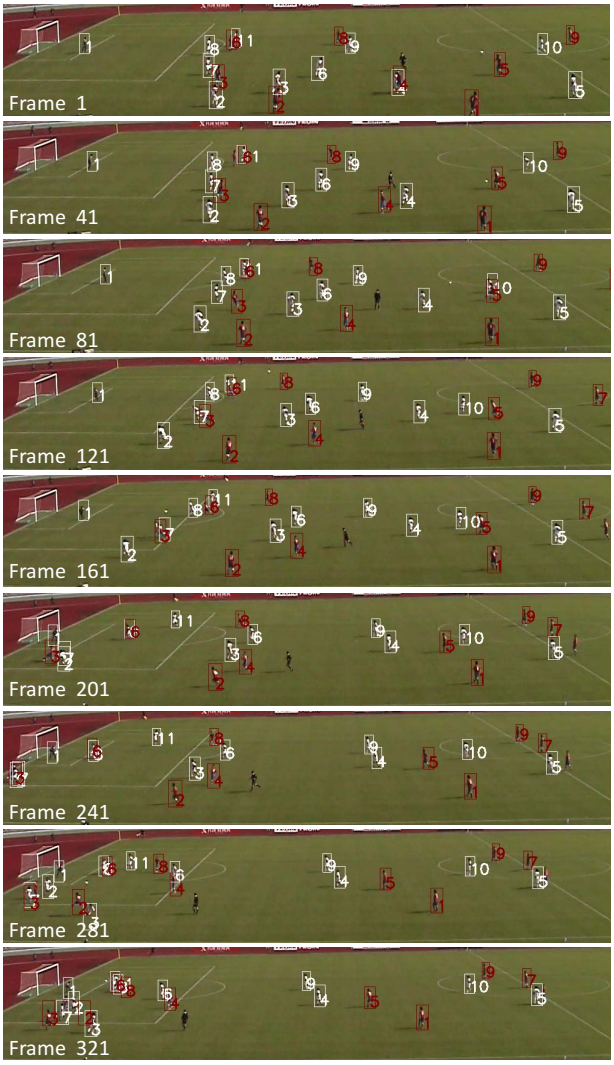

Figure 10. Tracking result

be recognized, there are many falsely recognized ones. They are high cheers, wind sounds, performance using musical instruments and so on. These failures are perhaps caused by their similar pitch. Therefore, we will devise a solution by eliminating the noise and increasing the number of HMM models trained by wind sounds and cheers other than whistle sounds.

Table V
THE ACCURACY OF WHISTLE SOUNDS RECOGNITION

	Rate (%)
Precision	66.67 (18/27)
Recall	100 (18/18)

3) Evaluation of Event Recognition: We evaluated the event recognition accuracy using the simulation video. In a soccer game, there are 8 events in all. In the experiment,

4 events among them (corner kick, throw-in, goal kick and free kick) were recognized. Two types of HMM models, players HMM and ball HMM, were trained using 30 scenes for each event respectively. 40 scenes (10 scenes for each event) in total were used as the test data.

Figure 11 shows the averaged recognition rate as a function of the combination weight α, which is employed in Eq (3), changing between 0 and 1. From Figure 11, the event recognition rate is highest (0.9) at $\alpha = 0.95$. Therefore, ball tracking is the most important factor for event detection and recognition as shown in Figure 11. However, since it is possible to detect and recognize events using only players movement even when a ball is lost, player tracking is also important.

Table VI shows the recognition rate of each event at α = 0.95. It turns out that all events are recognized almost correctly.

As a result, since the proposed system works well in the experiment for the simulation video, we will evaluate the proposed system in the actual video in a future.

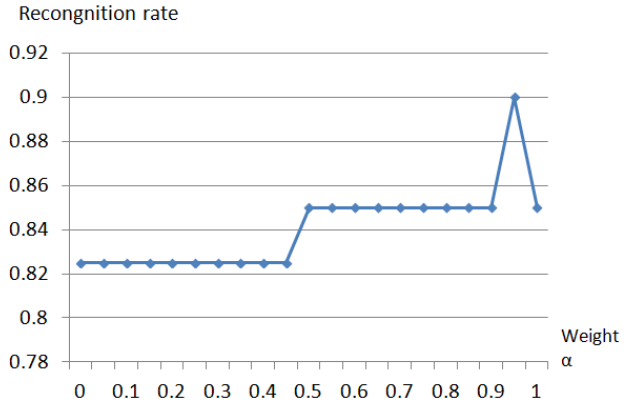

Figure 11. The averaged recognition rate as a function of the combination weight α

Table VI
THE RECOGNITION RATE OF EACH EVENT AT $\alpha = 0.95$

Event	Rate (%)
Corner kick	80 (8/10)
Throw-in	90 (9/10)
Goal kick	100 (10/10)
Free kick	90 (9/10)

VI. CONCLUSION

In this paper, we proposed a new method to detect and recognize events robustly in a soccer game. It was difficult to detect "free kick" and "throw in" because these events occurred anytime and anywhere in the game. In a soccer game, some event occurs when the referee blows a whistle or a ball is out of field. Therefore, we improved the detection accuracy of the events such as "free kick" and "throw in" by using these information when they occurred. Also, event recognition were performed by integration method of the results obtained using two types of HMMs : one is for players and the other is for a ball. As a result, the proposed system works well for the events such as free kick and throw-in in the experiment for the simulation video.

In a future, we will evaluate the proposed system for actual videos. Also, all 8 events including 4 events ("offside", "kickoff", "penalty kick" and "foul") will be evaluated. Since all events in a soccer game occur in the relationship between players and a ball, we will employ a new feature based on this relationship.

REFERENCES

[1] J. Ren, J. Orwell, G. A. Jones and M. Xu: "Tracking the soccer ball using multiple fixed cameras," Computer Vision and Image Understanding, Vol. 113, Issue 5, pp. 633-642, 2009.

[2] T. Misu A. Matsui, M. Naemura, M. Fujii and N. Yagi: " Distributed Particle Filtering for Multiocular Soccer-ball Tracking, " IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 937-940, 2007.

[3] J. Kim and T. Kim: " Soccer Ball Tracking using Dynamic Kalman Filter with Velocity Control, " Sixth International Conference on Computer Graphics, Imaging and Visualization, pp. 367-374, 2009.

[4] Y. Ariki, S. Kubota and M. Kumano: " Automatic Production System of Soccer Sports Video by Digital Camera Work Based on Situation Recognition, " IEEE international workshop on Multimedia Information Processing and Retrieval, pp.851-858, 2006.

[5] T. Nishino, T. Takiguchi, and Y. Ariki: "Event Recognition Using the 3 Dimentional Position Information of a ball in Monocular Image Sequence," MIRU, pp. 1269-1276, 2009.

[6] S. Motoi, T. Misu, Y. Nakada, T. Yazaki, G. Kobayashi, T. Matsumoto and N, Yagi: "Bayesian event detection for sports games with hidden Markov model," Pattern Analysis and Applications, pp. 59-72, 2012.

[7] T. Nishino, T. Takiguchi and Y. Ariki: "Situation Recognition Using 3D Positional Information of Ball from Monocular Soccer Image Sequence, " The 2009 International Conference on Multimedia, Information Technology and its Applications, pp. 109–112, 2009.

[8] M. Breitenstein, F. Reichin, B. Leibe, E. Koller-Meier and L. V. Gool: "Robust tracking-by-detection Using a Detector Confidence Particle Filter," The 12th IEEE ICCV, pp. 1515-1522, 2009-9.

[9] G. Zhu, C. Xu, Q. Huang and W. Gao: "Automatic multi-player detection and tracking in broadcast sports video using support vector machine and particle filter, " IEEE International Conference on Multimedia & Expo, pp. 1629-1632, 2006.

[10] Pascual J. Figueroa a, Neucimar J. Leite and Ricardo M.L. Barros: "Tracking soccer players aiming their kinematical motion analysis," Computer Vision and Image Understanding, pp. 122-135, 2005.

[11] T. Nishino, T. Takiguchi, and Y. Ariki: “Tracking of Multiple Soccer Players Using a 3D Particle Filter Based on Detector Confidence,” Advances in Computer Science and Engineering, Volume 6, Issue 1, pp. 93-104, 2011.

[12] M.J. Vapnik, “The Nature of Statistical Learning Theory”, Springer, Heidelberg, 2001.