Journal of Information Hiding and Multimedia Signal Processing ©2014 ISSN 2073-4212
Ubiquitous International Volume **5**, Number **2**, April **2014**

Robust Player Tracking for Broadcast Tennis Videos with Adaptive Kalman Filtering

Min-Yuan Fang, Chi-Kao Chang, Nai-Chung Yang, Chung-Ming Kuo*, Shih-Ku Guang

Department of Information Engineering
I-Shou University
No.1, Sec. 1, Syuecheng Road, Dashu Township 840, Kaohsiung, Taiwan, R.O.C.
*Corresponding author: kuocm@isu.edu.tw

Received February, 2013; revised September, 2013

Abstract. *Player detection and tracking play an important role for the content analysis of broadcast tennis videos. It is still a challenge because the player size is small and many noises and interference exist in a tennis court, which often results in a failure of detection. In addition, occlusion of players in double matches causes a failure of tracking. In this paper, we propose a robust technique of player tracking using an adaptive Kalman filtering. The parameters of the Kalman filter are dynamically adjusted according to the detection results of players. Experimental results indicate that the proposed method improves the success rate of player tracking significantly, especially for the upper players as well as for double matches.*
Keywords: Player detection, Player Tracking, broadcast tennis video, Kalman Filtering.

1. **Introduction.** In the past decade, broadcast programs of sport games are quite popular among millions of audiences in the world. A huge amount of broadcast sport videos are generated every day. Processing the huge video data becomes tedious works. Therefore, automatic content analysis for sport videos has received much attention recently. The analysis of sports video generates various valuable applications such as highlighting, summarization, indexing/retrieval, athlete's training and entertainment. In the past few years, significant content analysis has been performed to various kinds of sports such as soccer, tennis, baseball, American football, etc. [1-18,21].

Player tracking can provide very useful information for sport content analysis [22]. For example of tennis sports, events such as net approach, baseline rally and ace ball can be detected by referring players' position in the court. Players' tactic in the matches can be also discovered by players' trajectory. Players' tactic is the useful investigation to the competitor before the matches for athletes and trainers. Therefore, player tracking becomes one of most important issue for content analysis of sport videos.

Player detection and tracking for broadcast videos are much more difficult than real videos due to the following reasons [17]:

(a) Cameras are not stationary; they are zoomed and rotated and often follow the players.
(b) The background is frequently changed and players move randomly during the play.
(c) A player may be segmented into multiple regions because of the differences in the color of shorts, jerseys, and socks used.

(d) Court colors and textures change with different stadiums such as US Open, Wimbledon Open and French Open.
(e) Shadows cast by the players or other objects in the scene.
(f) Occlusions of players.

For tennis videos, a more challenge task is the detection and tracking of the players in the upper-half court not only the small size but also the noise interference. We will give a brief analysis for the issue in the next Section.

In this paper, we propose a robust player tracking method to address the problems mentioned above. It aims at upper player tracking; of course, it is also applied to the lower player. The method is mainly based on an adaptive Kalman filter, in which the parameters are adjusted dynamically according to the detection performance of players. In Section 2, we describe the proposed method which is mainly composed of courtline detection and filtering, player detection, and player tracking with adaptive Kalman filtering. The experimental results are described in Section 3. Finally, the conclusion is drawn in Section 4.

2. **Problem Analysis.** As shown in Figure 1, the camera, which is used to capture a court view, is often located behind the court. As a result, the whole court can be partitioned into upper-half and lower-half courts. We denote the player in upper-half court as upper player, and the player in lower-half court as lower player. Due to the camera's viewpoint, the objects in the upper-half court are much smaller than ones in the lower-half court.

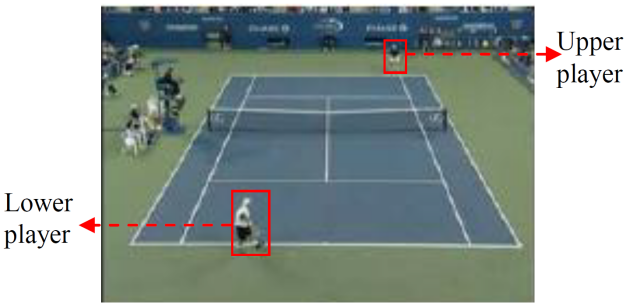


FIGURE 1. Lower player and upper players in a court view.

The backgrounds of the lower player are mainly the court fields with homogeneous colors. In addition, the image of the lower player is large enough. Therefore, it is not difficult to detect and track the player. The methods based on background subtraction or dominant color have been applied to segment the players successfully [19, 20, 23].

On the contrast, the detection/tracking of the upper players is a real challenge. The various objects such as commercial board, referee and staff are often mixed with the upper player. Because the background is complicated and varies over time frequently, it is difficult to generate an appropriate background dynamically; hence background subtraction hardly segments the upper player well. In addition, the upper player is quite small in size, which often results in poor detection of players. Moreover, for double matches, the occlusion of two players in a half court often causes tracking failure. Although, there are usually two cameras in a real tennis match, one behind each baseline, the upper-half court of one camera will become the lower-half court of the other camera. However, for most users, they cannot control the production of sport program, thus to develop an effective tracking technique is very desirable.

To further study the difficulty of upper player detection and tracking, we use an example to demonstrate the difference of resolution, i.e., size, between the two players in lower and upper half courts in a court image. As shown in FIGURE 2, the width of a real court is 36 ft, and for a typical player, the body height and width are assumed to be 6 ft and 1 ft and 6 inch, respectively. Assume the upper player and lower player stand on the baselines of the upper-half and lower-half courts, and the lengths of the two baselines in image space are L_1 pixels and L_2 pixels, particularly. The size of a player can then be calculated when court lines are detected. Given L_1 and L_2, we can estimate the widths of the upper player and lower player in image space, w_1 and w_2, by

$$\frac{(1*12+6)inch}{(36*12)inch} = \frac{w_1(pixels)}{L_1(pixels)} = \frac{w_2(pixels)}{L_2(pixels)} \tag{1}$$

After obtaining the widths of the two players in image space, we can further estimate the heights of the two players in image space by using the height and width of a player in real world. The estimation equation is denoted as

$$\frac{(6*12+6)inch}{(1*12+6)inch} = \frac{h_1(pixels)}{w_1(pixels)} = \frac{h_2(pixels)}{w_2(pixels)} \tag{2}$$

We employ the court line detection scheme presented in Section 2.1 to obtain the length of baselines, L_1 and L_2. Then we use Eq.(1) and Eq.(2) to estimate the sizes of the upper player and lower player. Table 1 lists the average results for three courts respectively including US Opens, French Opens and Wimbledon Opens. It indicates that the size of upper player is very small (52 × 14 pixels to 76 × 21 pixels), and it is about 32.4% 24.1% of the size of lower player. Therefore, the difficulty of the detection and tracking of the upper player increases significantly.

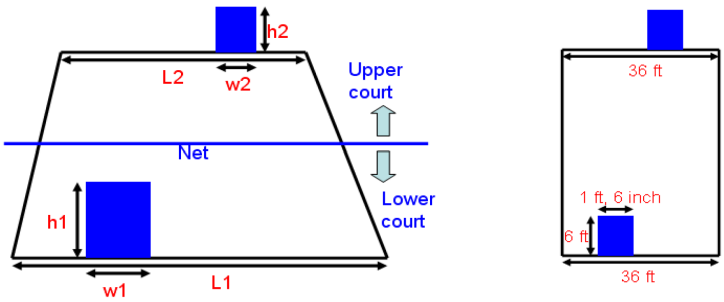


FIGURE 2. The relationship of court baseline length and player size: (A) In image space, (B) In real-word space.

The estimation of player's size in Table 1 is in an idea case. In practice, because the interference of the commercial board, or the color of a player is very similar to that of the playfield, the detectable player size is often much less than the estimation in Table 1. In the following, we demonstrate the difficulty of detection by an example. We use the detection method mentioned in Section 2 with a fixed detecting window of size 30×20 (in pixel), and the best and worst case for the detected player size is shown in Table 2. The detection results are also shown in FIGURE 3. Obviously, the detected player size varies dramatically, and even a very small one (11 to 27 pixels). Therefore, it is really a challenge for upper player detection and tracking.

TABLE 1. Estimate of player's size in lower-half court and upper-half court (in pixel).

	Lower- half court (in pixel)		Upper -half court (in pixel)		Size ratio of two players $(h_2 \times w_2)/(h_1 \times w_1)$
	Baseline length l1	Player's size $h_1 \times w_1$	Baseline length l2	Player's size $h_2 \times w_2$	
US Opens	632	104×29	314	52×14	24.1%
French Opens	549	90×25	316	52×14	32.4%
Wimbledon Opens	974	143×40	464	76×21	27.9%

TABLE 2. The detected upper player size in pixel.

	Maximum size	Percentage*	Minimum size	Percentage*	Average size	Average Percentage
US open 1	445	74.17%	25	4.17%	240	40.00%
US open 2	459	76.50%	27	4.50%	272	45.33%
US open 3	560	93.33%	188	31.33%	367	61.17%
French open	541	90.17%	124	20.67%	362	60.33%
Wimbledon Opens	384	64.00%	11	1.83%	116	19.33%

* The percentage is the ratio of detected player size and detecting window.

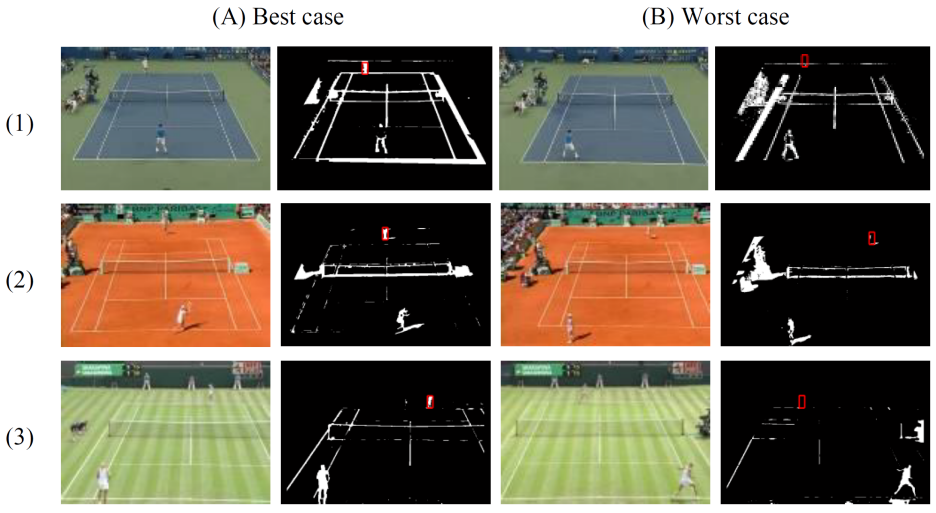


FIGURE 3. The detected players: (1) In us open, (2) In french open, (3) In wimbledon open.

3. **Proposed Method.** The flow chart of the proposed system is shown in FIGURE 4. To extract player objects, we first filter out playfield and court line using color features. Then, we detect player objects for the remaining image. Finally, the detection result is fed into an adaptive Kalman filter (KF) to estimate the player's position of each frame. The flow can be partitioned into two phases conceptually. In the detection phase, object extraction is performed for the first frame of the input video. For the subsequent frames, the tracking phase is conducted with the adaptive Kalman filter. Because the court line detection is not the main issue in this paper, so in Section 3.1 and 3.2, we briefly described some necessary processing and parameters that will be applied to player detection and tracking. For more detail, please refer to [19,24].

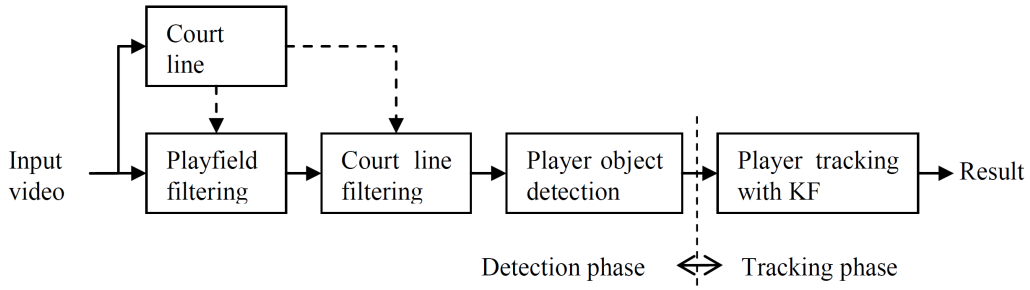


FIGURE 4. Flow chart of proposed method.

3.1. **Court line detection.** Court line information provides the import reference for the analysis of the court view. For example, side lines and base lines can be used to define the inside field and outside field, and a net line can be employed for the separation of the upper-half court and lower-half court. In our work, the court line detection is an essential preprocessing, which is helpful for the success of the following processes.

The flow chart of the court line detection is shown in FIGURE 5. We first transform the RGB color space into HSV space. The detection of court line is then performed in V channel. Through binarization and noise removing, we detect the candidate pixels which belong to the court lines. Radon transformation (RT) projects these candidates into peaks in Radon space. By searching these peaks, we can obtain the parameters of court lines, and equations of court lines can be calculated accordingly. FIGURE 6-8 illustrates the experimental results of each step in FIGURE 5 respectively. Apparently, the results are satisfactory and can be applied to subsequent phase. For more detail, please refer to [19].

3.2. **Playfield filtering and court line filtering.** For player tracking, we have to define the active region of player in a game. Thus, to find the playfield and define the active region of player according to court line is an important step for player detection and tracking. Playfield of a court can be characterized by the dominant colors of the court. The natural courts such as grass court and clay court have a single dominant color. However, the artificial court often has two dominant colors, one for the inside field and the other for the outside field, as shown in FIGURE 1. Thus, in our work, the playfield filtering considers two dominant colors for the two fields.

FIGURE 9(a) shows a schematic diagram of a tennis court. Five horizontal lines (h1-h5) and five vertical lines ($v1$-$v5$) construct a tennis court.

As shown in FIGURE 9(b), the inside field (F_{inside}) is enclosed by h_1, h_5, v_1 and v_5. Then, we define the outside field ($F_{outside}$) as the outward extended area from the inside court. Thus, Foutside is defined as the enclosed area by h_1', h_5', v_1' and v_5' but excluding

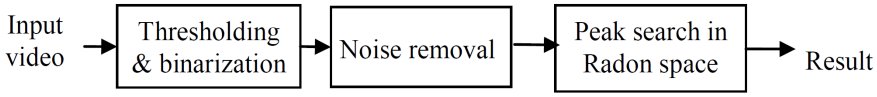
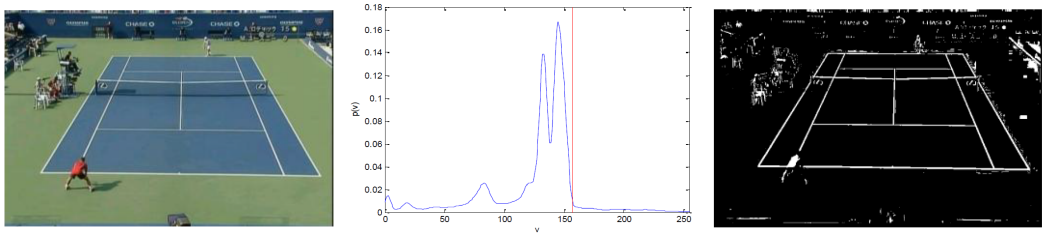
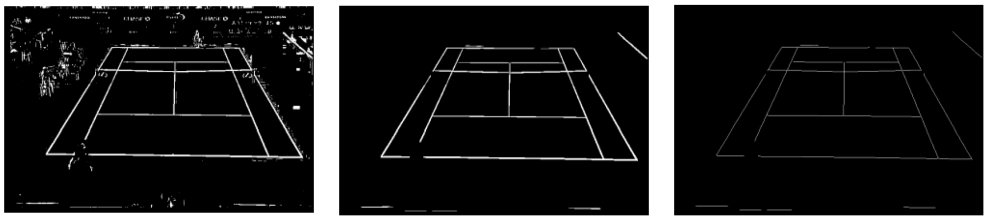


FIGURE 5. Flow chart of court line detection.



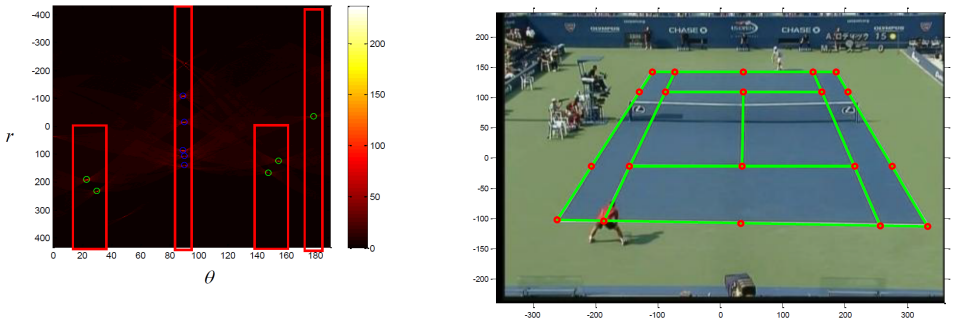
(a) (b) (c)

FIGURE 6. Thresholding and binarization of court lines: (A) Original image of court view, (B) Threshold determination, (C) The binarization result.



(a) (b) (c)

FIGURE 7. Illustration of noise removal and thinning.



(a) (b)

FIGURE 8. Court line detection: (A) Radon transformation of the thinning lines (B) Estimated court lines (redrawing line in green).

Finside. The left and right bounds of Foutside, v_1' and v_5', are respectively defined as the two lines outward extending 150 pixels (15/72 of image width) from v_1 and v_5. The bottom bound of $F_{outside}$, h_1', is defined as the line outward extending 100 pixels (15/72 of image height) from h_1.

Let the position of h_5' be at the corner of the wall behind the court. Here we estimate the position of h_5' using the gradient information of the upper image above h_5. Because the strongest edge happens in h_5', we use the maximum of the horizontal projection of the gradient image to detect the position of h_5', as illustrated in Eq.(3) and FIGURE 10.

$$y^{upper} = \text{vertical position of } h_5' = \arg_y \max\left(horproj(E)\right) \tag{3}$$

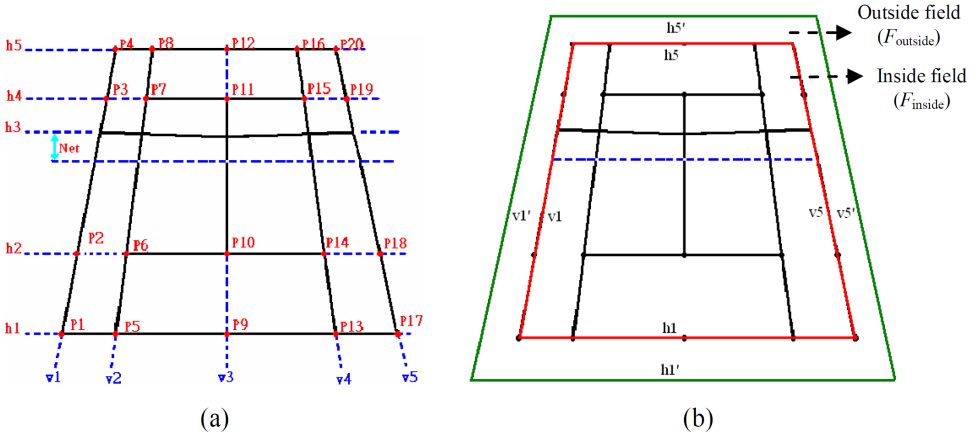


FIGURE 9. Schematic diagram of court lines.

where E and horproj(.) is denoted as the gradient image and horizontal projection, respectively. When the upper bound of the outside field is known, the $F_{outside}$ and F_{inside} can be easily determined, as shown in FIGURE 11.

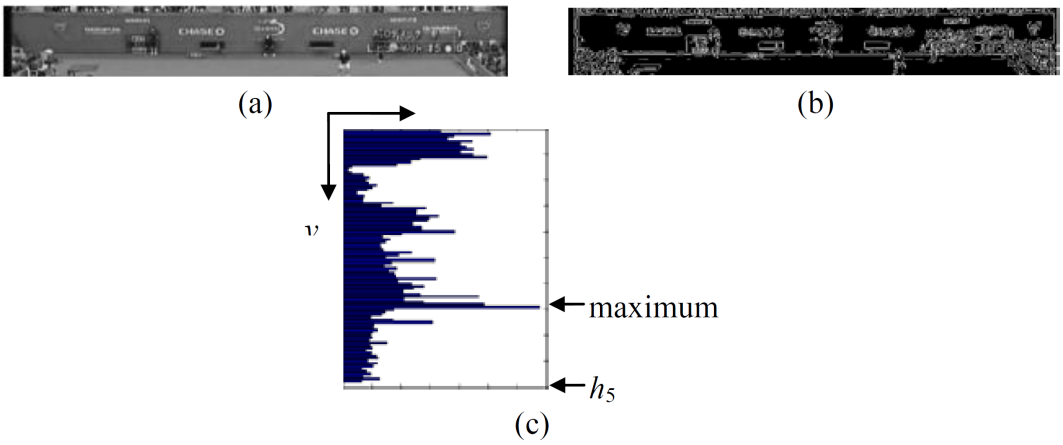


FIGURE 10. Detection of the upper bound of foutside: (A) Original image, (B) Gradient image of (A), and (C) Horizontal projection "horproj(.)" of (B).

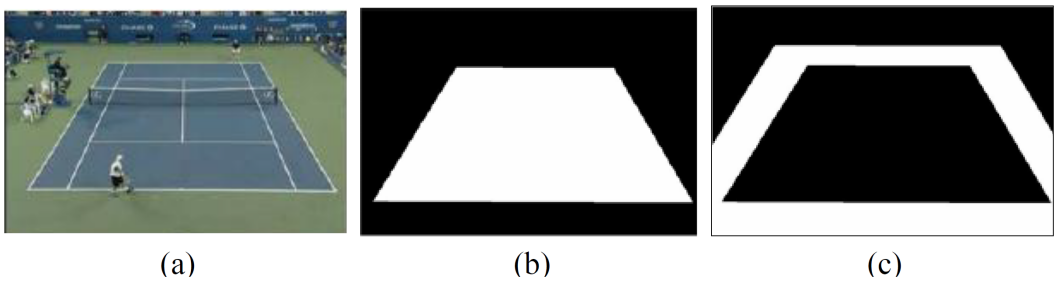


FIGURE 11. Boundary of finside and foutside (A) Original image, (B) Finside and (C) Foutside.

After inside and outside of playfield have been determined, the next step is to filter out the playfield pixels that are with dominant colors. Here we use color features to achieve this purpose. We first convert a RGB court image to the HSV color space. The hue value

and intensity value of a pixel at (x, y) are denoted by $hue(x, y)$ and $v(x, y)$ respectively. Then, we filter out the playfield's pixels using dominant colors within the outside and inside fields, and yield non-dominant color image (B_{NDC}) using Eq.(4).

$$B_{\mathrm{NDC}}(x,y) = \begin{cases} 0, \ if\ |hue(x,y) - \mu_{\mathrm{Hue}}| < \alpha\sigma^2_{\mathrm{Hue}} \text{ and } |v(x,y) - \mu_{\mathrm{Value}}| < \alpha\sigma^2_{\mathrm{Value}} \\ 1, \ otherwise. \end{cases}, \quad (4)$$

where μ_{Hue} and σ_{Hue} respectively represent the mean and variance of the inside or outside fields for hue component. Similarly, $\mu_{\{rmValue}$ and σ_{Value} are for intensity component. It is noted that Eq. (4) is only applied to the pixels within F_{inside} and F_{outside}. As a result, B_{NDC} filter out the playfield and contains only player objects, court lines, and other objects such as the net, as shown in FIGURE 12(a).

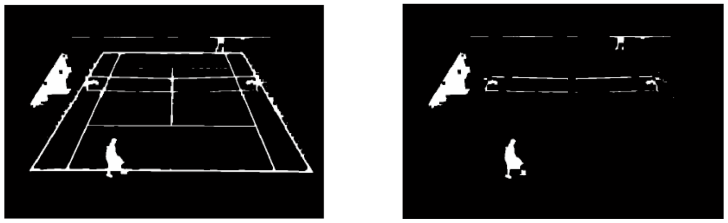


FIGURE 12. (A) B_{NDC}, (B) After court line filtering.

In order to detect player, we should remove court line first. We denote the court line as B_{CL}, as shown in FIGURE 7. The B_{CL} is operated by morphological dilation, and then subtract it from B_{NDC} as in Eq. (5). Finally, we can obtain a binary image containing candidates of player objects, denoted by B_{can}, as shown in FIGURE 12(b).

$$B_{\mathrm{can}} = B_{\mathrm{NDC}} - (B_{\mathrm{CL}} \oplus SE) \quad (5)$$

where SE is the square-shaped structure element of $n \times n$ matrix and $\oplus$ is the dilation operator.

3.3. **Player object detection.** This subsection describes how to detect (search) the player objects from the image B_{can}. To improve the detection accuracy and efficiency, we define a smaller initial search area using a priori knowledge. In general, when a player prepares to serve a ball, and he/she has to stand in the fixed range behind the baseline. In a single (match), the other player often stands behind the baseline in the opposite side and prepares to hit the ball back.

Given court line and playfield information, as shown in FIGURE 13, and the knowledge stated above, we can restrict the initial search area. For the upper-half court, the initial search area is defined as a rectangle box below the baseline line, h_5. Because the broadcasting style is different for different games, the court might extend to the outside of a frame. Thus, the search area is defined as follows.

Assume that X and Y respectively denote the horizontal and vertical coordinates of the intersection of court lines. The left and right bounds of the initial search area are defined as,

$$f^{upper}_{left} = \begin{cases} X_{P3}, \text{if } X_{P3} > 0 \\ 0, \text{ otherwise} \end{cases}, f^{upper}_{right} = \begin{cases} X_{P19}, \text{if } X_{P19} < framewidth \\ framewidth, \text{ otherwise} \end{cases} \quad (6)$$

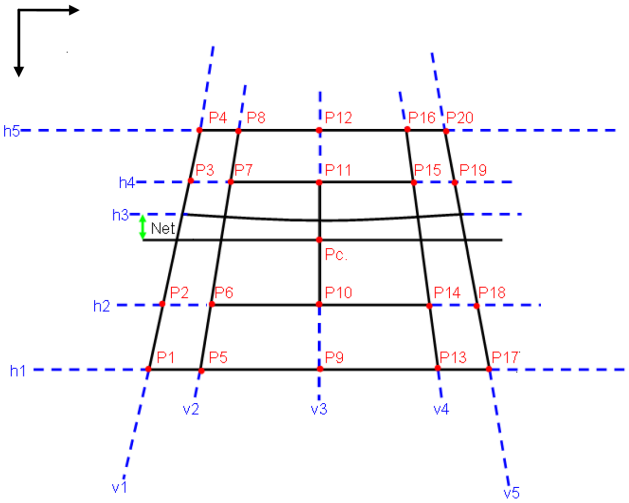


FIGURE 13. The depiction of court line.

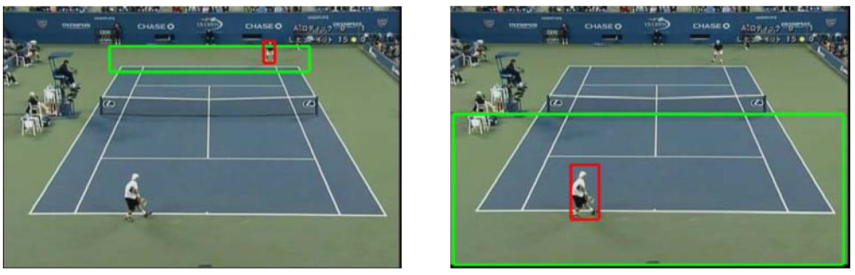
The upper and lower bounds are

$$f_{left}^{upper} = \begin{cases} y_{upper}, \text{if } y_{upper} > 0 \\ 0, \text{ otherwise} \end{cases}, \quad f_{down}^{upper} = \begin{cases} [max(Y_{P4}, Y_{P20}) + 10], \text{ if}[max(Y_{P4}, Y_{P20}) + 10] > 0 \\ 0, \text{ otherwise} \end{cases} \tag{7}$$

Because the noises in lower-half court are much less than those in upper-half court, the initial search area of the lower-half court can be defined as follows

$$f_{left}^{lower} = 0, \ f_{right}^{lower} = framewidth, \ f_{up}^{lower} = Y_{Pc}, \ f_{down}^{lower} = frameheight \tag{8}$$

where P_c is the center edge of net. The initial search areas for upper-half and lower-half courts are shown in FIGURE 14.



(a) (b)

FIGURE 14. Initial search area (green box) and the result of player detection (red box) for (A) Upper player, (B) Lower player.

The next step is to search the player object using a sliding window over the initial search area. The search is based on the binary image B_{can}. The sliding window is a small bounding mask with size of $\Delta W \times \Delta H$. In our work, the size of the bounding mask is 30 pixels×20 pixels. As shown in FIGURE 15, by shifting the sliding window pixel by pixel from left to right and top to bottom, we can calculate the object areas of all windows. The player's position is the location where the sliding window B_{st} generates the maximal object area; i.e.,

$$\text{Player}'\text{s position} = \arg\max_{s,t} \left(area_{bit-1}(B_{s,t})\right) \tag{9}$$

where areabit-1(.) denotes the total number of bit-1 pixels in the binary image B_{can}. Here, we use the bottom of the player window obtained in Eq.(9) to represent the player location. In our work, due to the noise interference and the limitation of resolution, the player extraction in playfield is not easy. For small or unimportant shadow, it will not affect the results of player extraction and tracking. Unlike the issue of posture or gesture recognition, which the precision of extracted player is necessary, we don't need to eliminate the shadow to reduce the interference. We will verify our viewpoint in simulation.

1. Using the property of 4-connectivity to find the bottom pixel of a player.
2. Check the bottom pixel of the player window. If it is not a player pixel, shift the player window one pixel upward; otherwise go to step 3.
3. Check the lower neighboring pixel of the bottom of the player window. If it is a player pixel, shift the player window one pixel downward and go to step 4; otherwise go to step 4.
4. Calculate the middle point of the lower bound of the player window as the representative position of player.

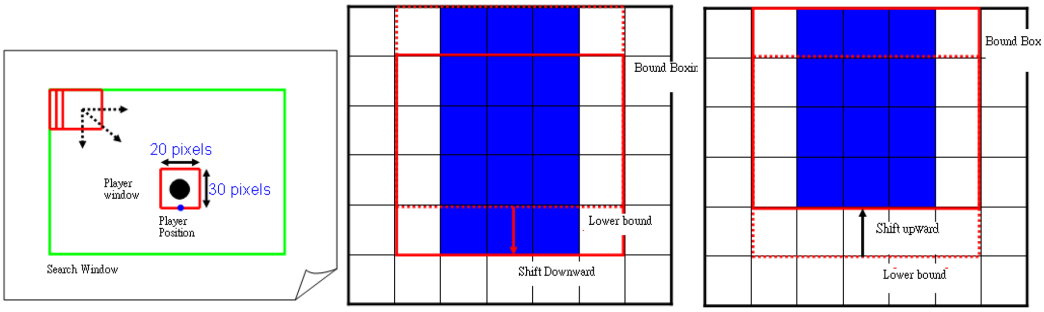


FIGURE 15. Searching player object.

The detection example for the upper player and the lower player is shown in FIGURE 14.

3.4. **Player's position estimation using Kalman filter.** After the player detection, we use a $\Delta W \times \Delta H$ bounding box to enclose the detected player. The next step is to estimate the player's position for the subsequent frames, which is the so-called tracking phase. We define a search window centered at the bounding box, as shown in FIGURE 16. According to the maximal movement of a player, we define the size of the search window to be $2 \times \Delta W$ width and $1.5 \times \Delta H$ height. Within the search window, a full search scheme sliders the bounding box to find a new position which generates maximal object area for each frame. The calculation of object area is similar to that defined in Eq. (9).

The upper player is quite small in size, and the background behind the upper player is rather complicated. As a result, the player detection process above can't obtain a complete player body. This results in the failure of the tracking using the above object search algorithm. In this work, we design an adaptive Kalman filter to solve the problem.

The system state model of Kalman filter for the player tracking is shown in Figure 17. The Kalman filter consists of the prediction process and the updating process. The position measurement and the occupation rate from the player object detection are fed into the updating process. Then, the occupation rate is used to predict the new position of the player object. The new position is applied to detect the player object in the next frame. The details are described in the following.

The state-space model of the Kalman filter is described as

$$\text{state model}: \ \mathbf{v}(k) = \mathbf{\Phi}(k-1)\mathbf{v}(k-1) + \mathbf{\Gamma}\mathbf{w}(k), \text{and} \tag{10}$$

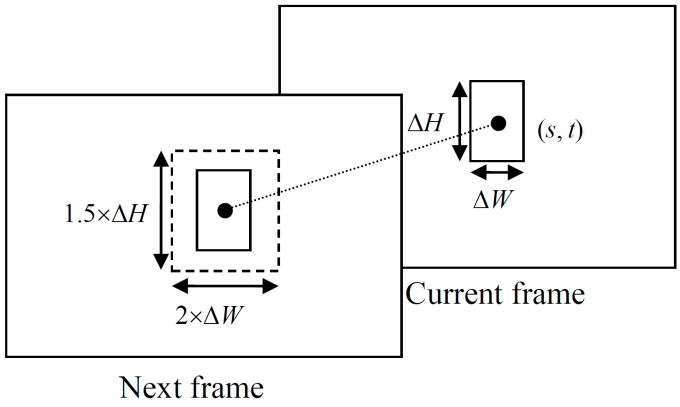


FIGURE 16. Search window centered in a bounding box.

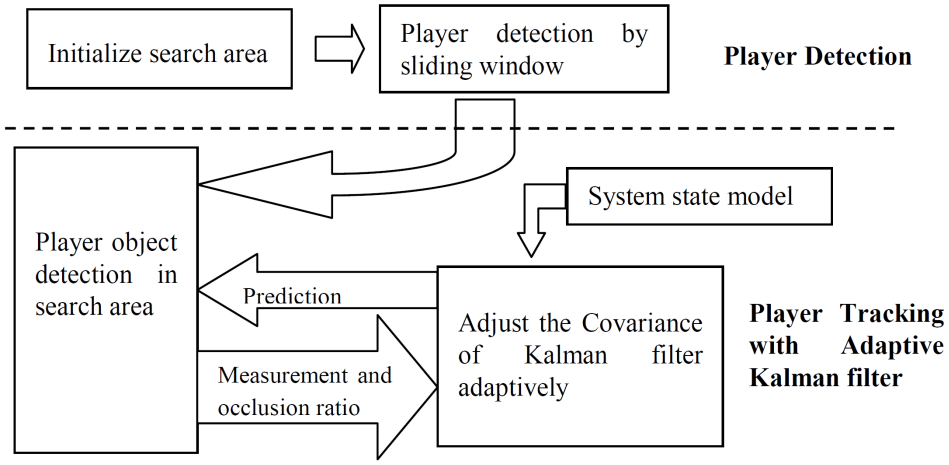


FIGURE 17. Flow chart of player tracking with adaptive kalman filter method.

$$\text{measurement model}: \ \mathbf{z}(k) = \mathrm{H}(k)\mathbf{v}(k) + \mathbf{e}(k) \tag{11}$$

where $\mathbf{\Phi}(k-1)$ and $\mathbf{H}(k)$ are the state transition matrix and measurement matrix, respectively. Assume the $\mathbf{w}(k)$ and $\mathbf{e}(k)$ are Gaussian noise with zero mean,; that is, $\mathbf{w}(k)$=N(0,Q(k)) and $\mathbf{e}(k)$=N(0,R(k)), where Q(k) and R($\mathbf{k}$) are process error covariance and measurement error covariance matrices, respectively.
Prediction process:

$$\text{state prediction}: \ \hat{\mathbf{v}}^{-}(k) = \mathbf{\Phi}(k-1)\hat{\mathbf{v}}^{+}(k-1) \tag{12}$$

$$\text{error covariance}: \ \mathbf{P}^{-}(k) = \mathbf{\Phi}(k-1)\mathbf{P}^{+}(k-1)\mathbf{\Phi}^{T}(k-1) + \Gamma\mathbf{Q}(k-1)\Gamma^{T} \tag{13}$$

Updating process:

$$\text{Kalman gain matrix}: \ \mathbf{K}(k) = \mathbf{P}^{-}(k)\mathbf{H}^{T}(k)[\mathbf{H}(k)\mathbf{P}^{-}(k)\mathbf{H}^{T}(k) + \mathrm{R}(k)]^{-1} \tag{14}$$

$$\text{state updating}: \ \hat{\mathbf{v}}^{+}(k) = \hat{\mathbf{v}}^{-}(k) + \mathbf{K}(k)[\mathbf{z}(k) - \mathbf{H}(k)\hat{\mathbf{v}}^{-}(k)] \tag{15}$$

$$\text{error covariance updating}: \ \mathbf{P}^{+}(k) = [\mathbf{I} - \mathrm{K}(k)\mathbf{H}(k)]\mathbf{P}^{-}(k) \tag{16}$$

The $\mathbf{P}(k)$ is the error covariance matrix associated with the state estimate of $v(k)$. It is defined as

$$\mathbf{P}(k) = E[(v(k) - \hat{v}(k))(v(k) - \hat{v}(k))^{T}] \tag{17}$$

This matrix provides a statistical measure of the uncertainty in $v(k)$. The superscripts " - " and " + " denote "before" and "after" measurement, respectively.

We derive the state-space mode for player tracking in the following. Because the interval between the two consecutive frames is very short, let us assume that the moving speed of a player (moving object) is constant. In addition, the x-direction and y-direction position of the tracking object are assumed mutually independent. Based on the assumptions, we can formulate the x-direction or y-direction position of the player using three subsequent frames k^{th}, $(k-1)^{\text{th}}$ and $(k-2)^{\text{th}}$ frame as

$$d(k) = d(k-1) + s(k) * 1 + w(k) \tag{18}$$

where $d(k)$ denotes the position of the player and $w(k)$ is the process noise. The speed of a player $s(k)$ can be estimated as the position difference of $(k-1)^{\text{th}}$ and $(k-2)^{\text{th}}$ frames by

$$s(k) = d(k-1) - d(k-2) \tag{19}$$

Thus, Eq.(21) becomes

$$d(k) = 2d(k-1) - d(k-2) + w(k) \tag{20}$$

However, the interference in upper player is significant especially for doubles match. The interference is mainly due to two possible reasons in the following. (a). The detectable player size is very small, and the non-court objects such as commercial board, score board, line judge, and logo of TV channel appear frequently; (b) The relative motion of players in double match is more complicated due to fast movement of players, and the two players may occlude each other. Therefore, the uncertainty of both measurement and prediction are very non-stationary. In order to reduce the interference and sensitivity, we refine the motion model such that it is more robust in noisy environment. The velocity is defined within longer time duration to avoid the randomness due to the fast movement and imperfect player detection. Thus, instead of Eq.(19), the velocity is calculated by $(k-1)^{\text{th}}$ frame and $(k-n)^{\text{th}}$ frame as

$$s(k) = \frac{1}{n-1}\left[d(k-1) - d(k-n)\right] \tag{21}$$

The velocity can be viewed as a motion trend, which effectively reduces the sensitivity. Consequently, our proposed state model of Kalman filter can be represented as

$$\begin{gathered}\mathbf{v}(k) = \mathbf{\Phi}\mathbf{v}(k-1) + \mathbf{\Gamma}\mathbf{w}(k) = \begin{bmatrix} 1+\frac{1}{n-1} & -\frac{1}{n-1} \\ 1 & 0 \end{bmatrix}\begin{bmatrix} d(k-1) \\ d(k-n) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w(k), \\ \text{where } \mathbf{v}(k) = \begin{bmatrix} d(k) \\ d(k-1) \end{bmatrix},\ \mathbf{v}(k-1) = \begin{bmatrix} d(k-1) \\ d(k-n) \end{bmatrix}, \\ \mathbf{\Phi} = \begin{bmatrix} 1+\frac{1}{n-1} & -\frac{1}{n-1} \\ 1 & 0 \end{bmatrix} \text{ and } \mathbf{\Gamma} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\end{gathered} \tag{22}$$

The measurement model can be represented as

$$\begin{gathered}\mathbf{z}(k) = \mathbf{H}(k)\mathbf{v}(k) + \mathbf{e}(k) = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} d(k) \\ d(k-1) \end{bmatrix} + e(k), \\ \text{where } \mathbf{z}(k) = d(k), \mathbf{v}(k) = \begin{bmatrix} d(k) \\ d(k-1) \end{bmatrix} \text{ and } \mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix}.\end{gathered} \tag{23}$$

It is noted that the above model equations can be applied to estimate either horizontal (x-direction) coordinate or vertical (y-direction) coordinate.

In our work, the n is set to be 2 for single match, 5 for double match, respectively.

3.5. **Adaptive Kalman filtering.** After constructing the motion model and achieving the measurement with player detection, we can apply the adaptive Kalman filtering to track the player in video sequences. The system state model in the adaptive Kalman filtering is derived from motion model which is used in the prediction step. The adaptive Kalman filter means that the parameters of Kalman filter are adjusted frame by frame automatically. Since the state model is corresponding to a linear space invariant system, all model parameters are thus given except the uncertainty of state model and measurement model, $\mathbf{Q}(k)$ and $\mathbf{R}(k)$, respectively. Therefore, if we can carefully adjust the $\mathbf{Q}$ and $\mathbf{R}(k)$, the better tracking performance can be achieved. In the following, we describe how to calculate the filter parameters of $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ automatically. As in Eq.(15), the Kalman gain, $\mathbf{K}(k)$, can be simply viewed as inversely proportional to $\mathbf{R}(k)$. If $\mathbf{R}(k)$ is greater, then $\mathbf{K}(k)$ is smaller, which means that the measurement, $\mathbf{z}(k)$, is less important for state updating, as indicated in Eq.(15). Meanwhile, the current estimation result should trust the prior estimation, $\hat{\mathbf{v}}^{-}(k)$, more, so we must decrease $\mathbf{Q}(k-1)$. On the other hand, when $\mathbf{R}(k)$ is smaller and $\mathbf{K}(k)$ is greater, $\mathbf{z}(k)$ is more important. In this case, the estimation result should trust $\mathbf{z}(k)$ more, so we increase $\mathbf{Q}(k-1)$.

In our work, we apply the coverage ratio of the detected player object to adjust $\mathbf{Q}(k-1)$ and $\mathbf{R}(k)$ dynamically. In player object detection, as in Eq.(9), the player's position is the location where the sliding window B_{st} generates the maximal object area. The coverage ratio is defined as the area of the detected object compared to the area of the bounding window as

$$\alpha(k) = \frac{area(\text{detected player})}{area(\text{bounding window})} \tag{24}$$

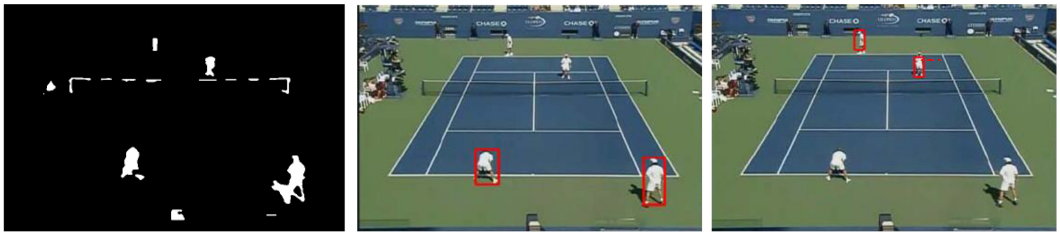
If the ratio is large, then the measurement is reliable. We can decrease the R(k) and increase Q(k) accordingly. Finally, Q($k-1$) and R(k) are simply defined as

$$Q(k-1) = \alpha(k) \text{ and } R(k) = 1 - \alpha(k) \tag{25}$$

For the detection of the upper player, because the moving player is often corrupted by the background noises such as commercial boards and side umpires, the detection of the player may be degraded seriously. Thus the size of the upper player detected becomes small; i.e., the coverage ratio is small. In such case, the proposed adaptive Kalman filter trusts the prediction much more than measurement; therefore it can reduce the effect of the unreliable measurement, and improve tracking accuracy significantly.

3.6. **Player tracking for doubles.** In a doubles match, two players are required to track in either of two half courts. As same as singles, the first step is to filter out playfield pixels and court line pixels to obtain player candidates image, B_{can}, as described in Subsection 3.2. Then, we search the two player objects in B_{can} which generate two largest numbers of bit-1 pixels inside the bounding box of players. In order to detect the players compactly, we use the 8-connectivity property to group the detected player pixels in the bounding box, and label the two players objects with red marks (1) and (2) shown in FIGURE 18(b) and (18(c). FIGURE 18(a) shows B_{can} obtained by playfield and court line filtering. FIGURE 18(b)-(c) respectively show the detection results for upper players and for lower players.

In the tracking phase, we use two Kalman filters to estimate two players' position in either of two half courts. Each Kalman filter executes the single-player tracking independently. Thus, we rewrite Eqs. (12)-(16) into the following



(a) (b) (c)

FIGURE 18. Player detection in a double: (A) bcan after filtering, (B) Lower players, (C) Upper players.

$$\begin{aligned}
&\text{Preditionstep}: \begin{cases} \hat{\mathbf{v}}^{-}(k) = \mathbf{\Phi}(k-1)\hat{\mathbf{v}}_i^{+}(k-1) \\ \mathbf{P}_i^{-}(k) = \mathbf{\Phi}(k-1)\mathbf{P}_i^{+}(k-1)\mathbf{\Phi}^T(k-1) + \Gamma\mathbf{Q}_i(k-1)\Gamma^T, \end{cases} \\
&\text{Updationstep}: \begin{cases} \mathbf{K}_i(k) = \mathbf{P}_i^{-}(k)\mathbf{H}^T(k)[\mathbf{H}(k)\mathbf{P}_i^{-}(k)\mathbf{H}^T(k) + \mathrm{R}_i(k)]^{-1} \\ \hat{\mathbf{v}}_i^{+}(k) = \hat{\mathbf{v}}_i^{-}(k) + \mathbf{K}_i(k)[\mathbf{z}_i(k) - \mathbf{H}(k)\hat{\mathbf{v}}_i^{-}(k)] \\ \mathbf{P}_i^{+}(k) = [\mathbf{I} - \mathrm{K}_i(k)\mathbf{H}(k)]\mathbf{P}_i^{-}(k) \end{cases}
\end{aligned} \tag{26}$$

where the subscript means the i-th player, i.e. $i = 1$ or 2.

For doubles matches, occlusions of two players happen frequently in a half court. The occlusions often make the Kalman filters not successful to work such that tracking errors occur (miss or mistake of tracking). To address the problem, we propose a new mechanism for $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ adjustment in the two Kalman filters. For two players in upper court, we observe their position measurements $d_1(k)$ and $d_2(k)$, and calculate the distance between them. When two players' position approach each other, the assignment of $Q_i(k-1)$ and $R_i(k)$ is changed as shown in Eq.(27).

$$\begin{aligned}
&\text{if } |_{\mathrm{x}}d_i(k) - {}_{\mathrm{x}}d_2(k)| < 1.5 \cdot \Delta W \text{ and } |_{\mathrm{y}}d_i(k) - {}_{\mathrm{y}}d_2(k)| < 1.5 \cdot \Delta H \\
&\text{then } Q_i(k-1) = 0 \text{ and } R_1(k) = \infty \\
&\text{else } Q_i(k-1) = \alpha_i(k) \text{ and } R_i(k) = 1 - \alpha_i(k)
\end{aligned} \tag{27}$$

where the left subscripts of "x" and "y" mean "x-component" and "y-component" respectively. The size of the bounding box of a player is $\Delta W \times \Delta H$. When the positions of two objects are very close (the distances of x-direction and y-direction of two players are less than $1.5 \cdot \Delta W$ and $1.5 \cdot \Delta H$ respectively), the occlusion happens. In such case, the position measurements of two players are very unreliable. Instead of Kalman filtering, the estimation use only prediction, i.e. $Q_i(k-1) = 0$ and $R_i(k) = \infty$. Until two player leave each other far enough, the measurements information are referred again for the estimation, i.e., $Q_i(k-1) = \alpha_i$ and $R_i(k) = 1 - \alpha_i$. The new mechanism effectively reduce the interference of unreliable measurement, and improve the tracking accuracy significantly.

4. **Experimental Results.** In our experiments, we record several videos of tennis matches from broadcast channels which were taken place in US Opens, French Opens and Wimbledon Opens. We obtain the tennis videos containing three different kinds of playfields including artificial, red clay and grass courts. We manually edit the clips out of the whole match videos. All of clips are videos of rallying between two sides, i.e. players running and stroking. The experimental materials contain 54 clips of 20 seconds in average which come from 10 matches, including 48 clips for singles and 6 clips for doubles. The video format is MPEG-2, that is, image resolution is 720×480 and frame rate is 30 fps.

For singles, Table 3 lists tracking results of the proposed method without Kalman filtering (only the player object detection). The success rate is 77% in average. According to our observation, most cases of the tracking misses are caused by the complex background including non-playfield objects such as commercial board, scoreboard caption and side umpire. Using dominant colors of playfield can not filter out these objects completely. As a result, the player object detection possibly tracks to the non-playfield objects in the half-upper court. In addition, when the color of the player's clothes is similar to the color of the playfield, over filtering happens to the player object and makes the object area very small. This is another reason for the failure of the player object detection.

Our proposed Kalman filtering cooperates with the player object detection to improve the tracking performance. The success rate of the tracking is raised from 77% to 94% in average, as shown in Table 4; that is to say, the Kalman filtering obtains 17% improvement. The key point is that when the measurement accuracy decreases, the prediction automatically compensate the unreliability by adaptive Kalman gain adjustment; because the motion of player can be viewed as continuous and smooth, so that the prediction can always effectively correct the error of poor measurement.

Figure 19-21 show the sequential frames of the successful tracking with Kalman filtering which are selected from three clips of US Opens, French Opens and Wimbledon matches. Figure 22 shows two tracking misses without Kalman filtering, when the upper players (tracking targets) approach side umpires.

For doubles, besides the factors of the complex background, the occlusion of two players affects tracking performance significantly. Table 5 lists the tracking results of doubles without and with Kalman filtering. Without Kalman filtering, the tracking of doubles does not succeed when the occlusion happens. The tracking with Kalman filtering achieves 66.7% success rate.

Figure 23 shows the successful tracking with Kalman filtering for the video clip selected from US Opens match.

Table 3. Tracking result of singles without kalman filtering.

	# of video clip	# of success	# of miss	success rate
US Opens	35	28	7	80%
French Opens	7	6	1	85.7%
Wimbledon Opens	6	3	3	50%
Total	48	37	11	77%

Table 4. Tracking result of singles with kalman filtering.

	# of video clip	# of success	# of miss	success rate
US Opens	35	33	2	94.3%
French Opens	7	7	0	100%
Wimbledon Opens	6	5	1	83.3%
Total	48	45	3	94%

TABLE 5. Tracking result of doubles in us opens with and without kalman filtering.

	# of video clip	# of success	# of miss	success rate
without KF	6	0	6	0%
with KF	6	4	4	66.7%

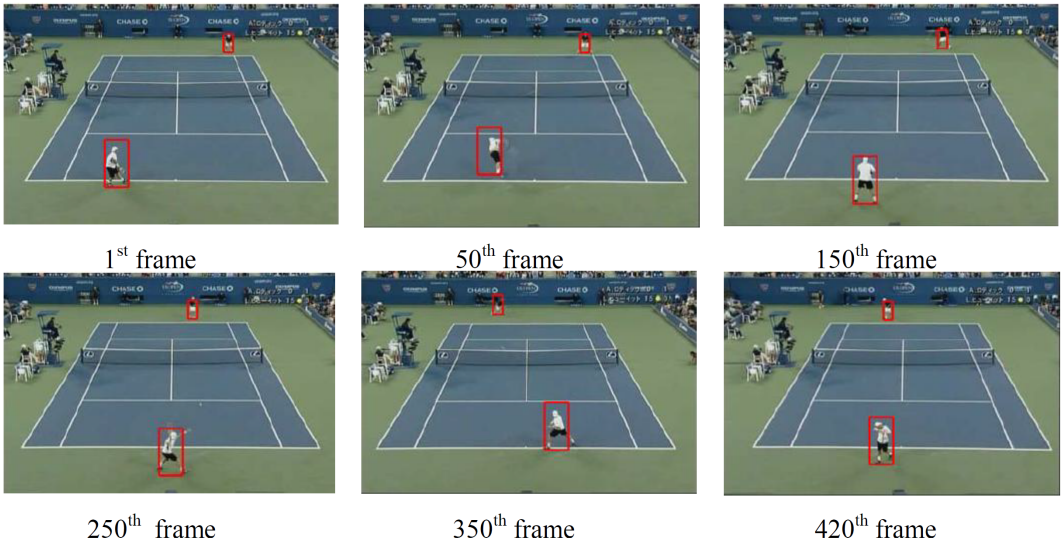


FIGURE 19. Tracking result in us opens with kf.

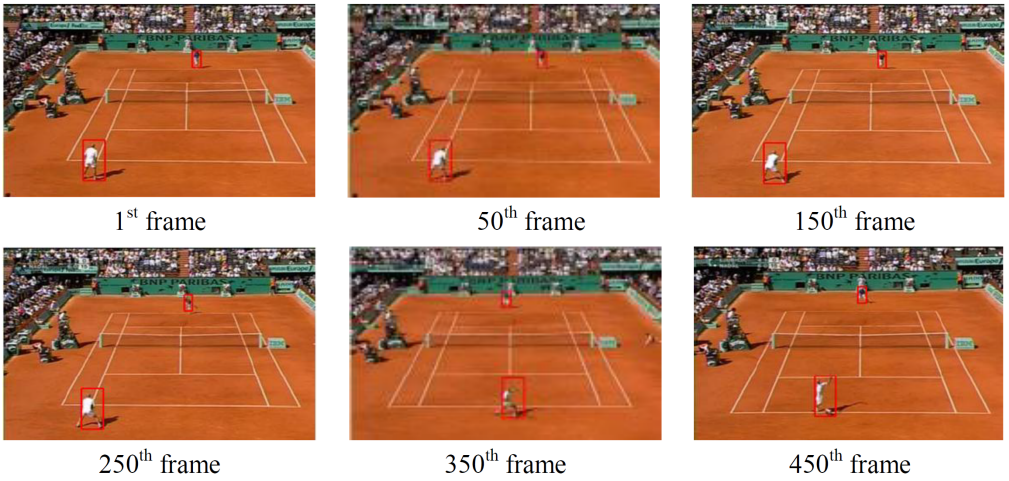


FIGURE 20. Tracking result in french opens with kf.

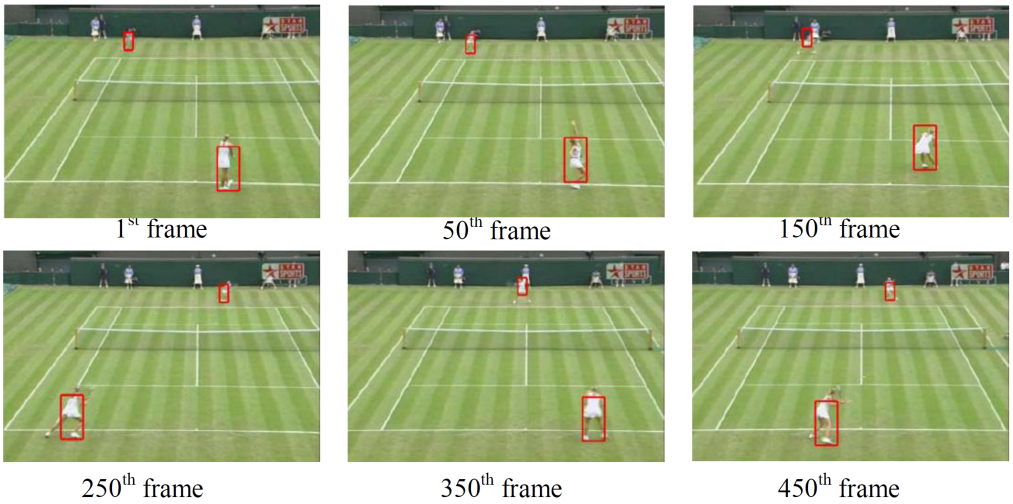


FIGURE 21. Tracking result in wimbledon opens with kf.

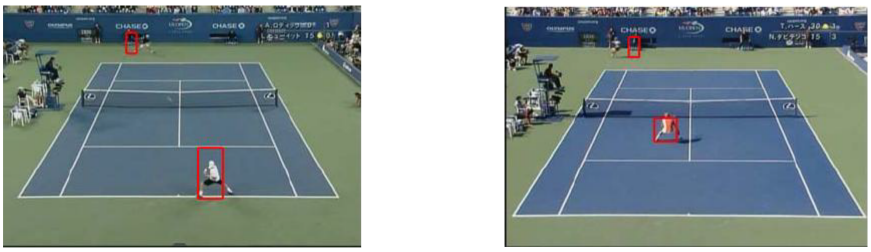


FIGURE 22. Two tracking misses in us opens (without kf).

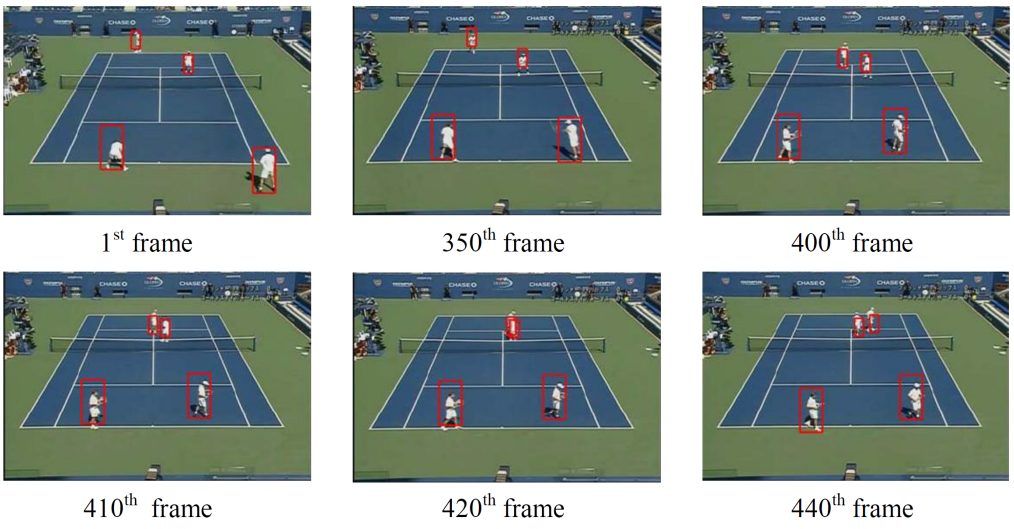


FIGURE 23. Tracking result in a double, us opens (with kf).

For more demonstration, FIGURE 24-25 show the player motion trajectory with and without Kalman filter in image plane and real-world space, respectively. Due to the imperfect player detection, the variation of bounding box is significant, and thus the abrupt jumps occur in the trajectories. However, the Kalman filter can effectively compensate the imperfect detection; therefore the trajectories are smoother.

Finally, for double match, the prediction model is modified to reduce the uncertainty of player motion. FIGURE 26-27 compare the tracking results with and without model

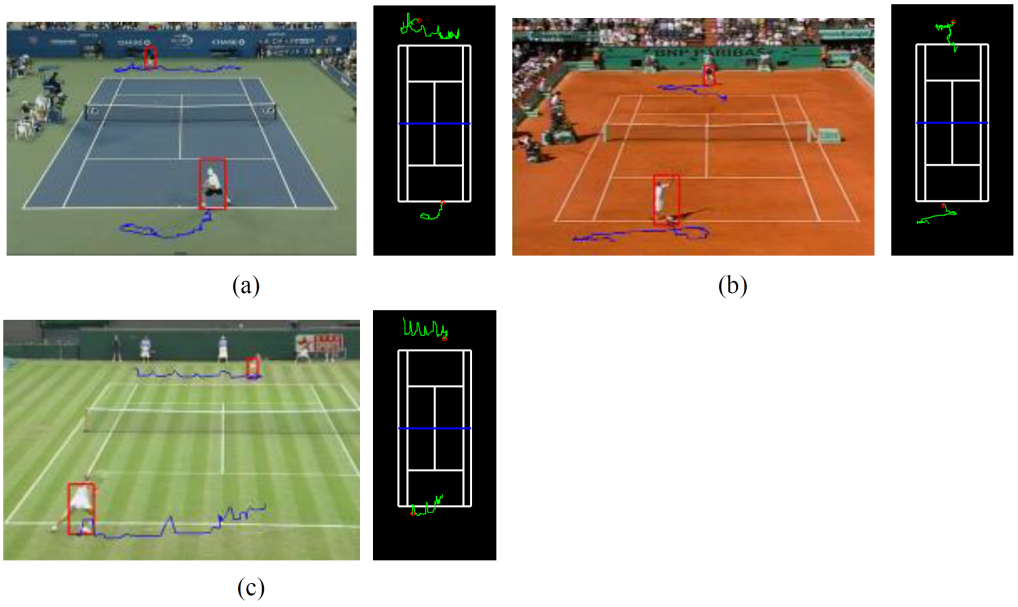


FIGURE 24. The Trajectories without kalman filter: (A) In us open, (B) In french open, (C) In wimbledon open.

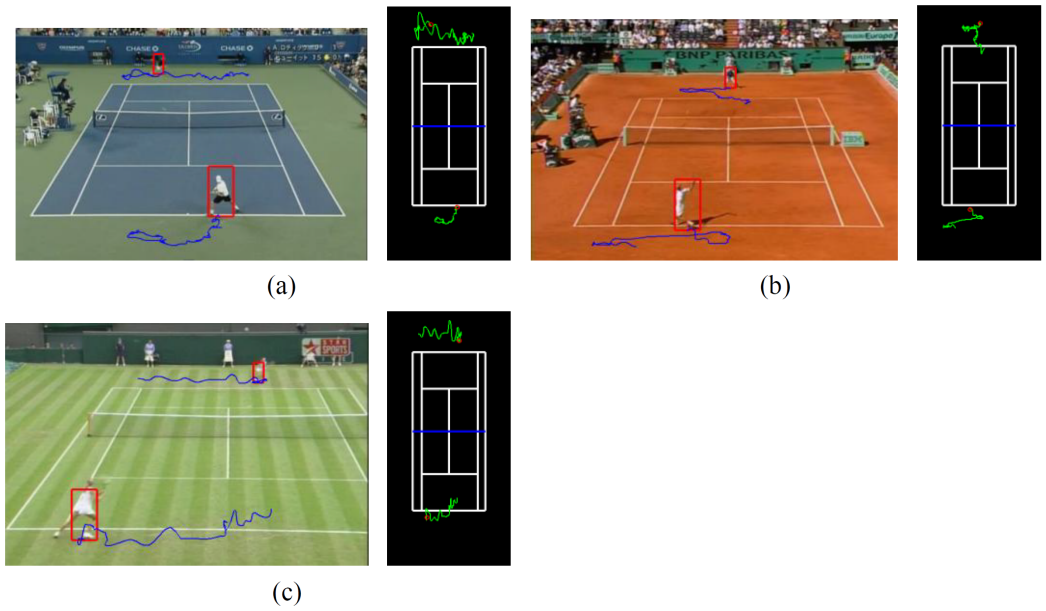


FIGURE 25. The trajectories with kalman filter: (A) In us open, (B) In french open, (C) In wimbledon open.

modification. Obviously, because the model modification effectively reduces randomness of player motion, it performs much better in tracking than the original model

5. **Conclusion.** In this paper, we have proposed a robust Kalman based player detection and tracking technique for broadcast tennis videos. In the detection phase, the playfield and court line are first filtered out from a court view. Then, the remaining is applied to detect player objects in a delimited search area. In the tracking phase, a bounding box containing the detected object (player) is used to search the position of the player in the next frame. The utilization of an adaptive Kalman filtering greatly corrects the detection

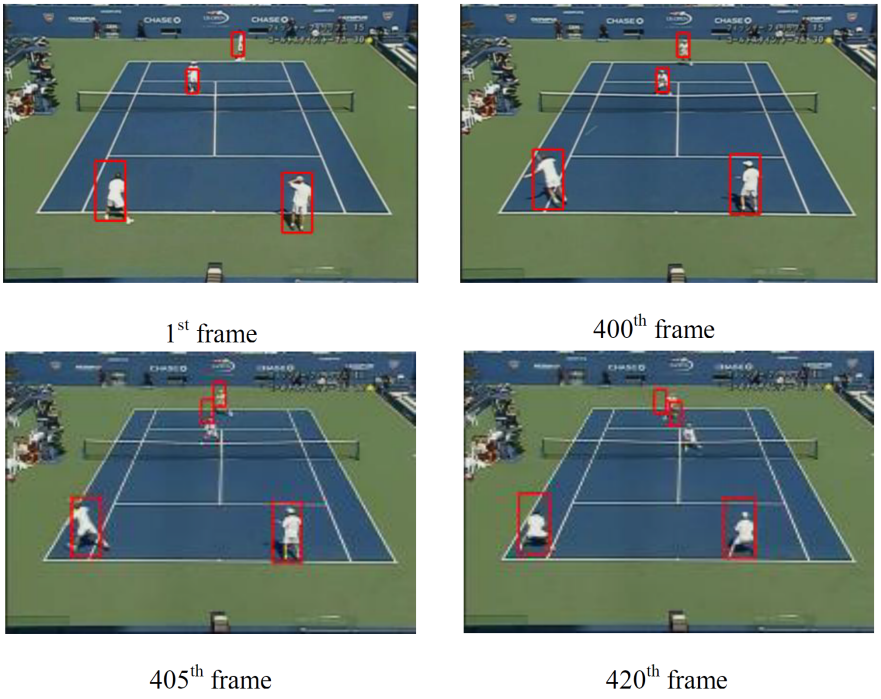


FIGURE 26. The trajectories with kalman filter: (A) In us open, (B) In french open, (C) In wimbledon open.

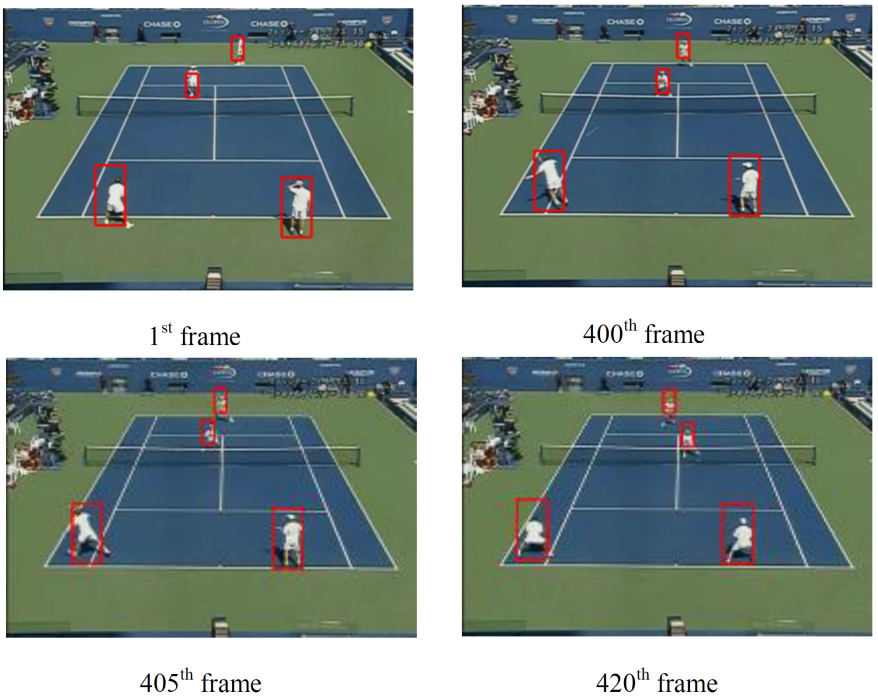


FIGURE 27. The trajectories with kalman filter: (A) In us open, (B) In french open, (C) In wimbledon open.

(measurement) errors and improves the tracking accuracy. Effective mechanisms for automatically adjusting parameters $\mathbf{R}(k)$ and $\mathbf{Q}(k)$ are developed based on the occupation ratio in the detection phase. The experiments indicate that the proposed method achieves

average success rate of 94% for singles and of 67.7% for doubles. The applications based on this result such as event detection and tactics analysis will be investigated in the future.

Acknowledgment. The authors would like to express their sincere thanks to the anonymous reviewers for their invaluable comments and suggestions. This work was supported by the National Science Counsel of Republic of China Granted NSC 101-2221-E-214-072.

REFERENCES

[1] A. Ekin, A. M. Tekalp, and R. Mehrotra, Automatic soccer video analysis and summarization, *IEEE Trans. Image Processing*, vol. 12, no. 7, pp. 796-806, 2003.

[2] D. Zhang, and S. F. Chang, Real-time view recognition and event detection for sports video, *Journal of Visual Communication and Image Representation*, vol. 15, no. 3, pp. 330-347, 2004.

[3] B. Li, J. H. Errico, H. Pao, and I. Sezan, Bridging the semantic gap in sports video retrieval and summarization, *Journal of Visual Communication and Image Representation*, vol. 15, no. 3, pp. 393-424, 2004.

[4] L. Xie, P. Xu, S. F. Chang, A. Divakaran, and H. Sun, Structure analysis of soccer video with domain knowledge and hidden Markov models, *Journal of Pattern Recognition Letters*, vol. 25, no. 7, pp. 767-775, 2004.

[5] R. Leonardi, P. Migliorati, and M. Prandini, Semantic indexing of soccer audio-visual sequences: a multimodal approach based on controlled Markov chains, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 634-643, 2004.

[6] Y. Gong, M. Han, W. Hua, and W. Xu, Maximum entropy model-based baseball highlight detection and classification, *Journal of Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 181-199, 2004.

[7] L. Y. Duan, M. Xu, Q. Tian, C. Xu, and J. S. Jin, A unified framework for semantic shot classification in sport video, *IEEE Trans. Multimedia*, vol. 7, no. 6, 2006, pp. 1066-1083, 2005.

[8] D. Sadlier, and N. O'Connor, Event detection in field sports video using audio-visual features and a support vector machine, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 15, no. 10, pp. 1225-1233, 2005.

[9] C. L. Huang, H. C. Shih, and C. Y. Chao, Semantic analysis of soccer video using dynamic Bayesian network, *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 749-760, 2006.

[10] J. L. Jian, M. H. Hung, C. H. Hsieh, and Y. Chang, Real-time scene classification for baseball videos, *Proc. of the 18th IPPR Conference on Computer Vision, Graphics and Image Processing*, pp. 115-122, 2005.

[11] Y. M. Su, and C. H. Hsieh, A novel caption extraction scheme for various sports captions, *Proc. of the 18th International Conference on Pattern Recognition*, pp. 1054-1057.

[12] Y. M. Su, and C. H. Hsieh, A novel model-based segmentation approach to extract caption contents on sports videos, *Proc of IEEE International Conference on Multimedia and Expo*, pp. 1829-1832, 2006.

[13] W. T. Chu, and J. L. Wu, Development of realistic applications based on explicit event detection in broadcasting baseball videos, *Proc. of the 12th International Multi-Media Modelling Conference*, pp. 12-19, 2006.

[14] J. Assfalg, and M. Bertini, Semantic annotation of soccer videos: automatic highlights identification, *Journal of Computer Vision and Image Understanding*, vol. 92, no. 2-3, pp. 285-305, 2003.

[15] G. Sudhir, J. C. M. Lee, and A. K. Jain, Automatic classification of tennis video for high-level content-based retrieval, *Proc. of IEEE International Workshop on Content-Based Access of Image and Video Database*, pp. 81-90, 1998.

[16] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, Personalized abstraction of broadcasted American football video by highlight selection, *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 575-586, 2004.

[17] V. Pallavi, J. Mukherjee, A. K. Majumdar, and S. Sural, Graph-based multiplayer detection and tracking in broadcast soccer videos, *IEEE Trans. Multimedia*, vol. 10, no. 5, pp. 794-805, 2008.

[18] M. H. Hung, and C. H. Hsieh, Event detection of broadcast baseball videos, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 12, pp. 1713-1726, 2008.

[19] Y. C. Jiang, C. H. Hsieh, C. M. Kuo, and M. H. Hung, Court line detection and reconstruction for broadcast tennis videos, *Proc. of the 19th IPPR Conference on Computer Vision, Graphics and Image Processing*, 2007.

[20] J. Han, D. Farin, and P. de With, Multi-level analysis of Sports Video Sequences, *Proc. of SPIE Conference on Multimedia Content Analysis, Management, and Retrieval*, 2006.

[21] S. W. Sun, W. H. Cheng, Y. L. Hung, I. Fan, C. Liu, J. Hung, C. K. Lin, and H. Y. Mark Liao, Who's who in a sports video? An individual level sports video indexing system, *Proc. of the IEEE International Conference on Multimedia and Expo*, pp. 937-942, 2012.

[22] M. C. Hu, M. H. Chang, J. L. Wu, and C. Lin, Robust camera calibration and player tracking in broadcast basketball video, *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 266-279, 1999.

[23] H. Ben Shitrit, J. Berclaz, F. Fleuret, and P. Fua, Tracking multiple people under global appearance constraints, *Proc. of the 2011 IEEE International Conference on Computer Vision*, pp. 137-144, 2011.

[24] M. Y. Fang, C. K. Chang, N. C. Yang, I. C Jou, C. M. Kuo, Robust player tracking and motion trajectory refinement for broadcast tennis videos, *Proc. of 2011 International Conference on Electric and Electronics*, pp. 9-18, 2011.