Robust Player Tracking and Motion Trajectory Refinement for Broadcast Tennis Videos

Min-Yuan Fang, Chi-Kao Chang, Nai-Chung Yang, I-Chang Jou,
and Chung-Ming Kuo*

Department of Information Engineering
I-Shou University
No. 1, Sec. 1, Syuecheng Rd., Dashu Township, Kaohsiung 840, Taiwan
kuocm@isu.edu.tw

Abstract. Player tracking and trajectory detection play an important role in 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, and therefore often results in a detection failure. In this paper, we propose a robust technique for player tracking and trajectory modification using an adaptive Kalman filtering. Experimental results indicate that the proposed method improves the successful rate of player tracking significantly, especially for the upper court players.

Keywords: player tracking, trajectory detection, Kalman filtering.

1 Introduction

In the past few years, significant content analysis has been performed to various kinds of sports such as soccer, tennis, baseball, American football, etc. [1-5]. The motion trajectory of player can provide very useful information for sport content analysis. For example, in tennis sports, events such as net play, baseline rally and ace ball can be detected by referring players' position in the court. Players' tactic in the matches can be also discovered from the players' trajectory. Therefore, player tracking becomes one of most important issue for content analysis of sport videos. For tennis videos, a more challenge task is the detection and tracking of the players on the upper-half court. As shown in Fig. 1(a), we denote the player on upper-half court as upper player, and the player on lower-half court as lower player. Due to the camera's viewpoint, the objects on the upper-half court are much smaller than that of lower-half court.

To explain the difficulty of upper player detection and tracking, we use an example to demonstrate the difference in resolution. If the lengths of the two baselines in image space are detected, we can use simple proportional relationship, as shown in

* Corresponding author.

M. Zhu (Ed.): Electrical Engineering and Control, LNEE 98, pp. 941–949.
springerlink.com

Fig.1 (b), to estimate the player size in both upper-half and lower-half court. In general, the size of upper player is very small (52×14 pixels to 76×21 pixels), and its size is about 32.4%~24.1% of the lower player. Therefore, the difficulty of the detection and tracking of the upper player increases significantly. In practice, because the interference of the commercial board or the color of a player would be very similar to that of the playfield, thus the practical detectable player size is often much less than the estimate. In this paper, we propose a robust player tracking and trajectory modification 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.

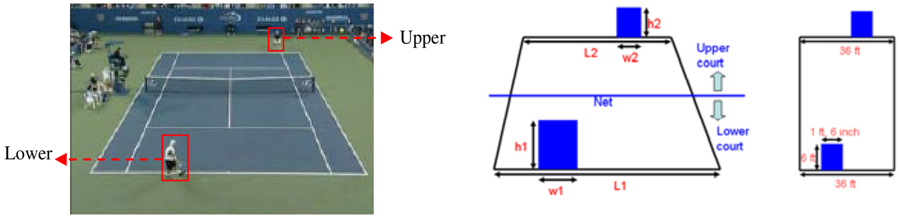


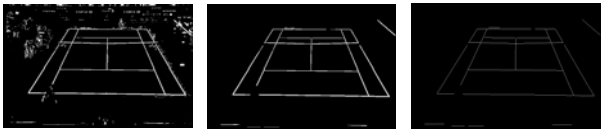
Fig. 1. Lower player and upper players in a court view and the relationship of court baseline length and player size: (a) in image space, (b) in real-world space.

2 Proposed Method

To extract player objects, we first filter out playfield and court line using color feature. Then, we detect players from the remaining image. Finally, the detection result is fed into an adaptive Kalman filter (KF) to estimate the player's position of each frame and modify the detected trajectory. The procedure can be partitioned into two phases conceptually. In the detection phase, object extraction is performed from the first frame of the input video. For the subsequent frames, the tracking phase is conducted with the adaptive Kalman filter.

2.1 Playfield Court Line Detection and Filtering

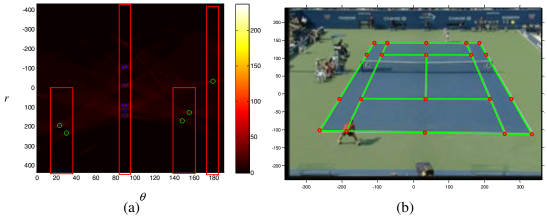
Court line information provides import reference for the analysis of the court view. Court lines are usually white. Thus, to detect white pixels is quite straightforward and efficient for the detection of court lines. We first transform the RGB color space into HSV space. The detection of court line is then performed in V channel. Through binarization and noise removing, 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. we detect the candidate pixels which belong to the court lines. Fig. 2 shows the results of court line detection and noise removing.



(a) (b) (c)

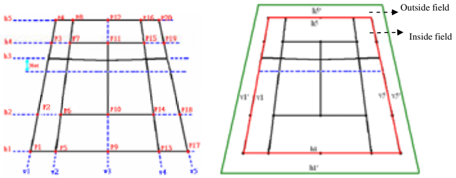
Fig. 2. Illustration of noise removal and thinning

The obtained image is then projected into Radon space using Radon transformation. It is obvious that the estimated lines (marked in green) match the original court lines very well. With the line equations, we can redraw court lines on the image of the court view, as shown in Fig. 4. Fig. 4(a) shows a schematic diagram of a tennis court. Five horizontal lines (h_1-h_5) and five vertical lines (v_1-v_5) construct a tennis court.



(a) (b)

Fig. 3. Court line detection: (a) Radon transformation of the thinning lines (b) estimated court lines (redrawing line in green)



(a) (b)

Fig. 4. Schematic diagram of court lines

To filter out the pixels that belong to the playfield, 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. The non-dominant color image (B_{NDC}) is defined as

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

where μ_{Hue} and σ_{Hue} represent the mean and variance of the inside or outside fields for hue component, respectively. Similarly, μ_{Value} and σ_{Value} are for intensity component. Note that Eq. (1) is only applied to the pixels within F_{inside} and F_{outside} as shown in Fig. 4(b). As a result, B_{NDC} filter out the playfield and reserves only player objects, court lines, and other objects such as the net, see Fig. 5(a). To detect player, we first widen court line images $\mathrm{B_{CL}}$ using morphological dilation operation. By subtracting the widened court lines from $\mathrm{B_{NDC}}$ as in Eq. (1), we obtain a binary image containing candidates of player objects, denoted by $\mathrm{B_{can}}$, as shown in Fig. 5(b).

$$B_{\mathrm{can}} = B_{\mathrm{NDC}} - (B_{\mathrm{CL}} \oplus SE)\,, \tag{2}$$

where SE is the square structure element of n×n matrix and ⊕ is the dilation operator.

2.2 Player Object Detection

Using the court line and playfield information, as shown in Fig. 3, and the knowledge stated above, we can restrict the initial search area. For the upper-half court, the initial searching area is defined as a rectangle area below the baseline line, h_5. Because the broadcasting style is different in each game, the court might extend to the outside of a frame. Thus, the searching area is defined as follows.

We denote X and Y as the horizontal and vertical coordinate of the intersection of court lines, respectively. The left and right bounds of the initial searching area are defined as,

$$f_{left}^{upper} = \begin{cases} X_{P3}, \text{ if } X_{P3} > 0 \\ 0, \text{ otherwise} \end{cases}, \qquad f_{right}^{upper} = \begin{cases} X_{P19}, \text{ if} X_{P19} < framewidth \\ framewidth, \text{ otherwise} \end{cases} \tag{3}$$

The upper and lower bounds are

$$f_{up}^{upper} = \begin{cases} y_{upper}, \text{ if } y_{upper} > 0 \\ 0, \text{ otherwise} \end{cases}, \qquad f_{down}^{upper} = \begin{cases} \left\lfloor \max\left(Y_{P4}, Y_{P20}\right) + 10 \right\rfloor, & \text{if} > 0 \\ 0, \text{ otherwise} \end{cases} \tag{4}$$

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

$$f_{left}^{lower} = 0, \quad f_{right}^{lower} = framewidth, \qquad f_{up}^{lower} = Y_{Pc}, \; f_{down}^{lower} = frameheight \tag{5}$$

where Pc is the center edge of net. The initial searching areas for upper-half and lower-half courts are shown in Fig. 6.

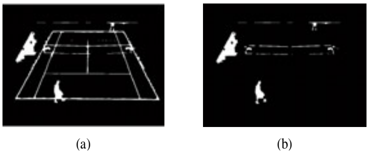


Fig. 5. (a) B_{NDC} , (b) Afte court line filtering.

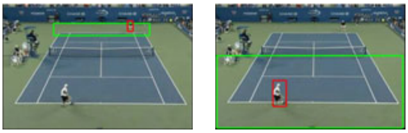


Fig. 6. Initial search area (green box) and the result of player detection (red box)

2.3 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. To estimate the player's position for the subsequent frames, we define a search window located on the center of the bounding box, as shown in Fig. 7. According to the maximal movement of a player, we define the size of the search window to be $2\times\Delta W$ in width and $1.4\times\Delta H$ in 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.

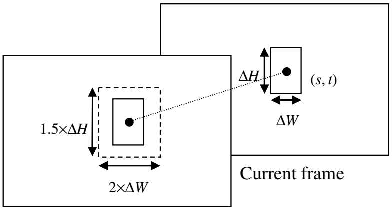


Fig. 7. Search window centered in a bounding box

The upper player is quite small in size, and the background behind the upper player is rather complicated. The player detection process is difficult to obtain a complete player's body. This results in failure of the tracking using the above object search algorithm. In this work, we design an adaptive Kalman filter to solve the problem.

Because the interval between the two consecutive frames is very short, let us assume that the moving speed of a player (moving object) keeps 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}, $(k-1)^{\mathrm{th}}$ and $(k-2)^{\mathrm{th}}$ frame as

$$d(k) = d(k-1) + s(k) * 1 + w(k) \tag{6}$$

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 from the position difference of $(k-1)^{\mathrm{th}}$ and $(k-2)^{\mathrm{th}}$ frames by

$$s(k) = d(k-1) - d(k-2) \tag{7}$$

Thus, Eq.(6) becomes

$$d(k) = 2d(k-1) - d(k-2) + w(k) \tag{8}$$

Consequently, our proposed state model of Kalman filter can be represented as

$$\begin{gathered}\mathbf{v}(k) = \mathbf{\Phi}\mathbf{v}(k-1) + \mathbf{\Gamma}\mathbf{w}(k) = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} d(k-1) \\ d(k-2) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w(k), \\ \text{where } \mathbf{v}(k) = \begin{bmatrix} d(k) \\ d(k-1) \end{bmatrix}, \ \mathbf{v}(k-1) = \begin{bmatrix} d(k-1) \\ d(k-2) \end{bmatrix}, \\ \mathbf{\Phi} = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix} \text{ and } \mathbf{\Gamma} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.\end{gathered} \tag{9}$$

The measurement model can be represented as

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{v}(k) + \mathbf{e}(k) = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} d(k) \\ d(k-1) \end{bmatrix} + e(k), \tag{10}$$

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

Assume the $\mathbf{w}(k)$ and $\mathbf{e}(k)$ are Gaussian noise with zero mean; that is, $\mathbf{w}(k)=N(0,\mathbf{Q}(k))$ and $\mathbf{e}(k)=N(0,\mathbf{R}(k))$, where Q(k) and R(k) are process error covariance and measurement error covariance matrices, respectively. We apply the occupation ratio of the detected player object to adjust $Q(k-1)$ and $R(k)$ dynamically. The occupation ratio is defined as the area of the detected object divided by the area of the bounding window as

$$\alpha(k) = \frac{area\left(\text{detected player}\right)}{area\left(\text{bounding window}\right)} \tag{11}$$

Finally, $Q(k-1)$ and $R(k)$ are simply defined as

$$Q(k-1) = \alpha(k) \text{ and } R(k) = 1 - \alpha(k) \tag{12}$$

The adaptive Kalman filter can reduces the effect of the unreliable measurement, and improves tracking accuracy significantly.

For motion trajectory of upper-court player, because the player size is too small and with too many noises interference, thus the trajectory is very unreasonable. As in Fig. (1), according to the relationship of image space and real-world space, we transfer the motion in image space to real-world space and use the following formula to modify the motion trajectory.

$$PT_{yupper}(t)=\left\{\begin{matrix} 5,\ when\ \ abs\left[\left(PT_{yupper}(t\text{-}1)\right)\text{-}\left(PT_{yupper}(t)\right)\right]\ <\ 5 \\ \frac{1}{10}PT_{yupper}(t)\ ,\ when\ \ 100>abs\left[\left(PT_{yupper}(t\text{-}1)\right)\text{-}\left(PT_{yupper}(t)\right)\right]\ \geq 5 \\ 100\ ,\ when\ \ abs\left[\left(PT_{yupper}(t\text{-}1)\right)\text{-}\left(PT_{yupper}(t)\right)\right]\ \geq\ 100 \end{matrix}\right\} \tag{13}$$

Because the player's position in vertical direction is more sensitive than that of horizontal, therefore the refinement is on vertical direction only. We use 5 cm and 100 cm as two threshold, and the trajectory is modified according to Eq.(13).

3 Experimental Results

In our experiments, we record several videos of tennis from broadcast channels, including US Opens, French Opens and Wimbledon Opens. All of clips are videos of rallying between two sides, i.e. player's running and stroking. The experimental materials contain 54 clips of 20 seconds in average, 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.

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 1; In other words, the Kalman filtering obtains 17% improvement.

Fig. 8 shows the sequential frames of the successful tracking with Kalman filtering and two tracking misses without Kalman filtering. The test clips are selected from US Opens. Fig. 9 shows the motion trajectory with and without refinement. Obviously, because the trajectory refinement effectively reduces randomness of the player's motion, it performs much reasonable in tracking than the original trajectory.

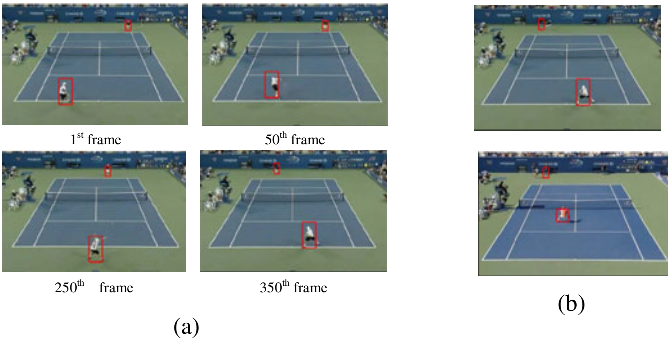
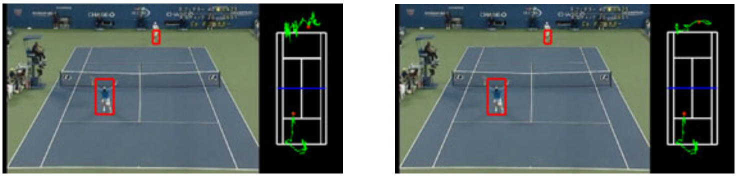

1st frame
50th frame
250th frame
350th frame
(a)
(b)

Fig. 8. Tracking result in US Opens (a) with KF (b) two tracking misses (without KF)



(a) (b)

Fig. 9. The motion trajectory (a) without trajectory refinement, (b) with trajectory refinement

Table 1. Tracking result of singles with(without) Kalman filtering

	A	B	C	success rate(%)
US Opens	35	33(28)	2(7)	94.3(80)
French Opens	7	7(6)	0(1)	100(85.7)
Wimbledon Opens	6	5(3)	1(3)	83.3(50)
Total	48	45(37)	3(11)	94(77)

A: # of video clip; B: # of success; C: # of miss.

4 Conclusion

In this paper, we have presented a robust technique, which is used for broadcast tennis videos, for player tracking and motion trajectory refinement. 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 searching 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 filter greatly corrects the detection (measurement) errors, and then improves the tracking accuracy. Effective mechanisms for automatically adjusting parameters $R(k)$ and $Q(k)$ are developed based on the occupation ratio in the detection phase. Finally, a refinement formula that smooths the motion trajectory is developed. The experimental results indicate that the proposed method achieves average success rate of 94% for tracking, and the motion trajectory is more reasonable than that of original trajectory. The applications based on this method such as event detection and tactics analysis will be investigated in the future.

Acknowledgments. This work was supported in part by National Science Counsel Granted NSC 98-2221-E-214-044-MY2 and NSC 99-2221-E-214-055.

References

1. Ekin, A., Tekalp, A.M., Mehrotra, R.: Automatic Soccer Video Analysis and Summarization. IEEE Trans. on Image Processing 12(7), 796–806 (2003)
2. Zhang, D., Chang, S.F.: Real-time view recognition and event detection for sports video. Journal of Visual Communication and Image Representation 15(3), 330–347 (2004)

3. Huang, C.L., Shih, H.C., Chao, C.Y.: Semantic analysis of soccer video using dynamic Bayesian network. IEEE Trans. on Multimedia 8(4), 749–760 (2006)
4. Xie, L., Xu, P., Chang, S.F., Divakaran, A., Sun, H.: Structure analysis of soccer video with domain knowledge and hidden Markov models. Pattern Recognition Letters 25(7), 767–775 (2004)
5. Leonardi, R., Migliorati, P., Prandini, M.: Semantic indexing of soccer audio-visual sequences: a multimodal approach based on controlled Markov chains. IEEE Trans. on Circuits Syst. Video Techn. 14(5), 634–643 (2004)
6. Pallavi, V., Mukherjee, J., Majumdar, A.K., Sural, S.: Graph-Based Multiplayer Detection and Tracking in Broadcast Soccer Videos. IEEE Transactions on Multimedia 10(5), 794–805 (2008)
7. Hung, M.H., Hsieh, C.H.: Event Detection of Broadcast Baseball Videos. IEEE Transactions on Circuits and Systems for Video Technology 18(12), 1713–1726 (2008)
8. Jiang, Y.C., Hsieh, C.H., Kuo, C.M., Hung, M.H.: Court line detection and reconstruction for broadcast tennis videos. In: Kalra, P.K., Peleg, S. (eds.) ICVGIP 2006. LNCS, vol. 4338, Springer, Heidelberg (2006)
9. Weng, S.K., Kuo, C.M., Tu, S.K.: Video object tracking using adaptive Kalman filter. Journal of Visual Communication and Image Representation 17(6), 1190–1208 (2006)