A Novel Algorithm For Shuttlecock Tracking

Zhishen Huang* and Wai-kuen Cham†
* The Chinese University of Hong Kong, Hong Kong
E-mail: zshuang@ee.cuhk.edu.hk Tel/Fax: +852-54910591
† The Chinese University of Hong Kong, Hong Kong
E-mail: wkcham@ee.cuhk.edu.hk Tel/Fax: +852-26035558

***Abstract*—Real-time accurate ball tracking in sport games is important for automated game analysis and augment reality display. It is very difficult because the ball is usually very small, with few features and large variance appearance, sudden motion change and occlusion occurs quite often as well as the background is noisy. Badminton is one of the fastest ball games in the world and so tracking a shuttlecock accurately is a challenging work.**

In this paper, we propose a novel real-time non-recursive tracking algorithm to addressing this issue by formulating the motion model and the correlation within scene context, which is different from standard tracking approaches. We build a multi-layer filters focusing on the rationality to eliminate false candidates step by step and choose the optimal candidate without previous results. Experiment results show that the proposed tracker can track the shuttlecocks in real-time with satisfying performance.

I. Introduction

Accurate ball tracking in sports is important to athletes, referees and audience. However, the size, velocity of the ball and prolonged occlusions make ball tracking challenging even for trained human observers. Shuttlecock, the fastest ball game in the world, is one of the most challenging work in tracking category because of many complex factors including occlusion and motion change, small scale with few visual model features, background clutter and severe object blur. Although object tracking has been investigated intensively in the past decade, it is still a challenging problem to design a robust visual tracking algorithm in real scenes.

Many tracking algorithms with significant performance are introduced in [14] and discriminative features are exploited to track the targets in the complex scenes with excellent performance. However, most of the tracking algorithms cannot be used for tracking a shuttlecock in badminton game in real time. The reason can be concluded as: i)The pose of a shuttlecock changes in the flight and so does its appearance. In addition, the shape of a shuttlecock in video frames is often a small white area occupying only a few pixels because of its small size. ii) Unlike a tennis ball, a shuttlecock often flies ten feet or higher above the floor. In order to capture its full trajectory, the background is often a complex scene containing audience and many objects. iii) The shuttlecock in a badminton game often travels at high speeds and appears as a blurred object in a frame. This leads to a large appearance distortion and makes the detection work more difficult. iv) Occlusion of the shuttlecock caused by players is much more frequent in a badminton game than a tennis game. Several algorithms are proposed for ball tracking. In [9] the author proposed a data association based algorithm to track tennis ball in tennis game video. The performance is good but the algorithm cannot be implemented in real time. In [3][4][5], trajectory based trackers are designed to track the a ball in sport games, but the algorithms are required to set the initial location of ball and they cannot cope with tracking failure.

(a) Frame I_d^n (b) Frame I_d^{n+1} (c) Frame I_d^{n+2}

Fig. 1: (a)(b)(c) are frame difference images generated by three consecutive frames. The location of the shuttlecock can be estimated from these frames.

Our goal is to design a real time shuttlecock tracking algorithm which is able to detect and track the ball independently in each frame. Before introducing the main idea of our algorithm, let's explore three images (Fig. 1 (a)(b)(c)) first. These images are generated from consecutive frames of game video with frame differencing and threshold operations. We find that people can easily tell where the ball is simply through these three images. How can we human do that through these binary images? We conclude that firstly, people can easily estimate the players and play ground in 3D space. Secondly, our eyes can capture the moving objects and estimate the possible moving trajectories by experience. Thirdly, the judgement is made based on the possible trajectories linking two players and the shape of the trajectory most likely to be a shuttlecock flying path. With the help of these information, people can infer that in the images showed in Fig.1, one player just hits the ball and the shuttlecock is flying from one player to another and thus easily tell where the shuttlecock is.

Starting from this particular idea, we propose a novel real-time non-recursive shuttlecock tracking algorithm focusing on the motion features and correlation with scene context. First, we explicitly model the correlation between shuttlecock and the information extract from scene context including the recover formulation from 2D to 3D space and shuttlecock flying trajectories in particular view. Second, we designed a multi-layer filters to eliminate the false candidate step by step. Third, a model function is defined to select the true candidate

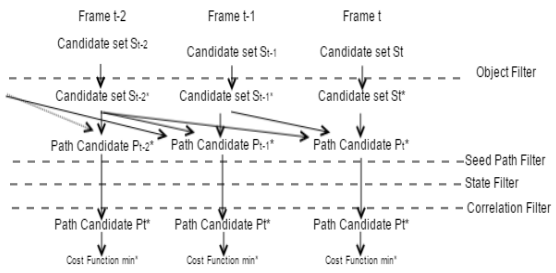


Fig. 2: Algorithm Structure

from candidates set passed through the multi-layer filters in each frame.

II. Proposed Algorithm

Our aim is to build a shuttlecock tracker which is able to detect the shuttlecock in real-time robustly and non-recursively with the help of scene context. The core idea is how to design the non-recursive structure and to find the correlation with scene context. In this section, we firstly explore the information hidden in the scene context which is helpful for tracking, then the algorithm is introduced to explain how to make it real-time and non-recursive.

A. Scene Context Correlation

A lot of information hidden in the scene context is helpful for tracking. We carefully explore the following clues which are helpful to locate the true candidates: 1. Shuttlecock has specific motion model. 2. Shuttlecock flying state can be estimated from player motion. 3. Strong correlation between ball trajectory and specific sources.

The coming question is how to extract and utilize these clues. Should these features be regarded as evidence to calculate probability for each candidate or should they be set as a criteria to eliminate false candidates? As the statement in the following sections, considering computing speed and approximate simulation, these features we extract in each frame are not precise enough to be set as condition for calculating the probability of each candidate, but they are good enough to be criteria to eliminate most of the false candidate with reasonable boundary condition.

B. Algorithm Structures

Our tracking algorithm structure is organized as Fig. 2 shows. At the beginning of each new frame, a set of connected components are generated by frame difference operation. All of these components are treat as ball candidates. In the second part, a multi-layer filter is designed to reduce the number of false candidates. Firstly an object level filter is introduced which simply eliminate false candidate based on the object size and transparency. A weak threshold is set to guarantee true candidate passing through. Combing with candidates set of previous frames, a new set of candidates defined as seed-path is formed and more attributes are extracted by the new combination. The following multi layer filter will eliminate the false candidates based on the attribute of seed-path including motion model, correlation with player state and location. This analysing method is non-recursive because the decision made in current frame does not depend on the true candidate information in previous frames. A seed-path description function is defined and the candidates with passed through the filters with minimum cost will be set as true candidate in current frame.

III. Scene Information Extraction

A. 3D Structure Reconstruction

Information utilizing 3D, most definitely offers much more distinct information for analysing. In badminton games, the advantages of 3D reconstruction can be classified as two aspects: recover 3D coordinate for pixels in image space and project 3D world information to image space. As the badminton playground has standard size and salient feature such as white board lines and clearly crossing points, it is not hard to get the 2D and 3D recover formulation from [11][13]. In our system, we adopted the pinhole camera model. By using this model, a scene view is formed by projecting 3D points into the image plane utilizing a perspective transformation. The corresponding relationship between points in the image (u, v) and 3D world coordinate (X, Y, Z) is formulated as:

$$s\,[u, v, 1]^T = M\,[X, Y, Z, 1]^T\,. \tag{1}$$

where $\mathbf{A}$ is the intrinsic matrix encompassing the focal length, image sensor format, and principal point.$[\mathbf{R}|\mathbf{t}]$ are the extrinsic parameters, which denote the coordinate system transformations and s is a scale factor. In the following part of this paper, we use the following equations to represent reconstruction and projecting relationship derived from (1):

$$\begin{cases} f([u, v]^T) = [X|_z, Y|_z, Z]^T \\ F([X, Y, Z]^T) = [u, v]^T \end{cases} \tag{2}$$

B. Trajectory Generation

In order to explore the correlation between possible trajectories and players, a set of flying trajectories which represent the possible flying path in current view must be pre-calculated. The shuttlecock flying trajectory is given by [10]:

$$y = \frac{v_t'^2}{g} \ln \left| \frac{\sin[\frac{v_t'}{v_{y_t}}(e^{\frac{gx}{v_t'^2}} - 1) + \tan^{-1}(\frac{v_t'}{v_{y_t}})]}{\sin[\tan^{-1}(\frac{v_t'}{v_{y_t}})]} \right| . \tag{3}$$

Setting the start speed as constant maximum speed (300km/h) with different initial angles, different shape of trajectories can be generated in a single phase. In this paper the number of trajectories we adopted is 12 which is precise enough to model different realistic trajectories. Each trajectory is represented as

$$T^o\,\{v_t, \theta_t, s_t\} \tag{4}$$

where v_t, θ_t and s_t are velocity, angle and state respectively in time t. Setting the trajectories in different plane phase, we can easily project the 3D trajectories into trajectories in image plane representation with (2). The new trajectories represent as:

$$T^{\gamma}\{v_t^{\Delta\gamma}, \theta_t^{\Delta\gamma}, s_t\ \} \tag{5}$$

C. Player Detection

Player detection is a easy task in badminton game as the players are isolated on the playground and constrained by size and location. A mean-shift cluster with bounding box constrains will get accurate result based on the components from frame difference. The state of player represents as:

$$PL\{b, l_2, l_3, \Delta\psi\}. \tag{6}$$

where b represents the bounding box in image space with center l_2, the location in 3D l_3 is estimated with (1) assuming the player always stepping on ground. $\Delta\psi$ is the normalized motion representation for each player calculated by the number of frame difference pixels. A larger value means player is moving pretty fast and smaller otherwise.

IV. Implement Details

A. Candidate Generation

The velocity of shuttlecock is always extreme fast which leads to severe motion blur. In addition, white is a common color. The small white shuttlecock can easily blend into complicated background. All of these factors lead to the candidate extraction work quite complicated. Therefore, to guarantee the true candidate can be extracted in every frame, the shuttlecock candidates are extracted simply through frame difference with connected-components labelling. All of the objects generated here are set as candidates.

B. Object Filter

Most of time the shuttlecock is blurred and blended into white background objects, thus the criteria that only keeps white object with regular shape does not make any sense here. In this filter, we use the color and transparency information to eliminate false candidates. The blurred shuttlecock in image can be regarded as a matting problem and we set the keeping criteria as components with good consistency.

C. Seed-path Filter

Seed-path is a small path formed by k candidates in k consecutive frames represents as:

$$sp\{\mathbf{p}, v, a, \theta, \Delta\theta\} \tag{7}$$

which satisfies:

$$\begin{cases} v = \|p_2 - p_1\|_2 / \Delta t - 0.5 a \Delta t \\ a = (\|p_3 - p_2\|_2 - \|p_2 - p_1\|_2)/\Delta t^2 \\ \theta = \angle(p_2 - p_1) \\ v + k_s a \in (0, v_{max}), 0 \le k_s \le k \\ a \in (a_{min}, a_{max}) \\ \Delta\theta < \theta_{thres} \end{cases} \tag{8}$$

where p means the center of candidates in image space, v, a and θ represent the velocity, acceleration and direction respectively extracted from the combination.

In this paper we set k as 4 for computing speed consideration and four components are good enough to calculate key feature parameters and cope with motion change and occlusion. The candidates set changes to seed-path instead of isolated objects in the following filters. Any seed-path which does not satisfy the model is discarded in current filter layer.

D. State Filter

Observing from the video, we conclude that the motion of the player has a great inference on trajectory state. The filter designed here is to eliminate the false candidate which is not able to occur under player state P_i in (4). For example, it is impossible for the seed-path to be with a large velocity when player stop moving or with extreme small velocity when player moving in a hurry. Here we construct a ratio representation to figure out the relationship between player and trajectories:

$$\sum_i PL(\Delta\psi)_i \propto \sigma v^{\beta} e^{-l^{\eta}}. \tag{9}$$

where σ is a scale in different view, v is the velocity in seed-path and l is the Euclidean distance between seed-path and player and β , η are adjusting parameters. Seed-path without reasonable ratio will be eliminated.

E. Correlation Filter

It is obvious that there is no chance for a seed-path to be true candidate if the location of the matched trajectory is far from players. In correlation filter, the seed-path is fitted into the trajectory to check whether the trajectory is between players. Trajectory matching is formulated as:

$$\arg\min_{i,t}\ \omega_1 \left\| T_i^{\gamma}(v_t^{\Delta\gamma}) - sp(v) \right\|_2 + \omega_2 \left\| T_i^{\gamma}(\theta_t^{\Delta\gamma}) - sp(\theta) \right\|_2 \tag{10}$$

The correlation between trajectory and players can be represented as:

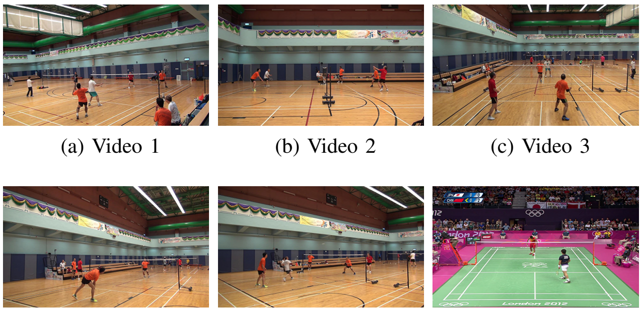
$$\begin{aligned} \text{min distance } &= d_1 + d_2 < \text{threshold distance} \\ s.t.\ &\text{cross point above } z \text{ plane.} \\ &\text{highest point above net} \end{aligned} \tag{11}$$

Mathematically calculated in Euclidean space with the following equation, z_{thres} represents error tolerance and z_{net} is the height of court net.

$$\begin{aligned} \min\ d_{tj} &= \left\| f([u_i, v_i]^T) - l_3^m \right\|_2 + \left\| f([u_j, v_j]^T) - l_3^n \right\|_2 \\ s.t.\ & z|_{f([u_i,v_i]^T)} + z_{thres} > 0. \\ & z|_{(u_i,v_i)(u_j,v_j)} + z_{\text{thres}} - z_{\text{net}} > 0 \end{aligned} \tag{12}$$

F. Final Decision

A model is set to select a better one as true candidates or make the judgement that there is no shuttlecock in current frame. A good model should have good consistency and here we choose the candidate with best model measurement. According to [12], MLESAC is proposed to give an accurate measurement but involves estimation of mixture parameters of



(a) Video 1 (b) Video 2 (c) Video 3

(d) Video 4 (e) Video 5 (f) Video 6

Fig. 3: Screen shot of 6 videos

a likelihood function. The function is complicated and computationally expensive thus not suitable for real-time processing. In this implementation, the following cost function is adopted:

$$\arg\min_{i} C = \sum_{i} \rho(p^{i}_{est})). \quad (13)$$

where

$$\rho(p^{i}_{est}) = \begin{cases} d^2(\widehat{p}_{est}, p_i) \ if \ d(\widehat{p}_{est}, p_i) < d_{th} \\ d^2_{th} \ otherwise \end{cases} \quad (14)$$

and p is the candidate location in image, d^2 represents the distance in Euclidean space with estimated location $\widehat{p}_k$ from seed-path model. A smaller C indicates a better model. The candidate passed through all filters with minimum cost smaller than the keeping threshold will be regarded as true candidate. For the situation that no seed-path passing through filters with small cost C, the decision will be made as no shuttlecock in current frame.

V. Experiments

The testing dataset consists of 6 video sequences. Each is about 10 minutes long. Fig. 3 summarises these sequences which were captured at different angles, resolution , frame rate different type of players. We manually labelled the true shuttlecock location in each frame. The tracking algorithm performance shows in Table I.

We set 4 parameters to evaluate the performance: True and False Positive(TP, FP) represent the detection success and failure rate when shuttlecock is flying. True and False Negative(TN, FN) represent the success rate to justify whether shuttlecock is flying. Analysing from the performance results, we can see that the proposed tracking algorithm can track the shuttlecock efficiently in different views. The tracking failures are mainly caused by the situation when shuttlecock merged into players or moving objects in the background. The algorithm is implemented in c++ without code optimization. The proposed tracker runs at 350 frames per second on an i7 machine.

TABLE I: Algorithm Performance

	TP	FP	TN	FN
Video 1	88.6%	11.4%	73.7%	26.3%
Video 2	92.7%	7.3%	81.2%	18.8%
Video 3	86.8%	13.2%	76.5%	23.5%
Video 4	89.1%	10.9%	77.7%	22.3%
Video 5	86.9%	13.1%	79.9%	20.1%
Video 6	93.2%	6.8%	85.7%	14.3%

VI. Conclusions

In this paper, we present a novel real-time tracking method for shuttlecock tracking in badminton game. The comprehensive experiments on six badminton game videos demonstrate that the proposed approach can successfully detect and track the shuttlecock in real-time based on the proposed multi-filter model and correlation with scene context scheme.

References

[1] Y. Seo, S. Choi, H. Kim, K. Hong, “Where are the ball and players? Soccer game analysis with color based tracking and image mosaic,” *in Proc. ICIAP (1997)*, pp. 196203.

[2] T.DOrazio,N.Ancona, G. Cicirelli,M. Nitti, “A ball detection algorithm for real soccer image sequences,” *in Proc. of 16th International Conference on Pattern Recognition*, vol. 1 (2002), pp. 210213 doi:10.1109/ICPR.2002.1044654

[3] X. Yu, H.W. Leong, C. Xu, Q. Tian, “Trajectory-based ball detection and tracking in broadcast soccer video,” *IEEE Trans. Multimed.* 8(6), 11641178 (2006)

[4] H.T. Chen, H.S. Chen, M.H. Hsiao, W.J. Tsai, S.Y. Lee, “A trajectory-based ball tracking framework with visual enrichment for broadcast baseball videos,” *J. Inf. Sci. Eng.* 24, 143157 (2008)

[5] H.T. Chen, M.C. Tien, Y.W. Chen, W.J. Tsai, S.Y. Lee, “Physics-based ball tracking and 3D trajectory reconstruction with applications to shooting location estimation in basketball video,” *Elsevier Sci. J. Vis. Commun. Image Represent.* 20, 204216 (2009)

[6] H. Shum, T. Komura, “A spatiotemporal approach to extract the 3D trajectory of the baseball from a single view video se-quence,” *in IEEE International Conference on Multimedia and Expo* (2004), pp. 15831586

[7] D. Liang, Y. Liu, Q. Huang, W. Gao, “A scheme for ball detection and tracking in broadcast soccer video,” vol. 3767 (Springer, Berlin, 2005), pp. 864875

[8] Lepetit, Vincent, Ali Shahrokni, and Pascal Fua. “Robust data association for online application,” *Computer Vision and Pattern Recognition*, IEEE Computer Society Conference on. Vol. 1. IEEE, 2003.

[9] Yan, F., Kostin, A., Christmas, W., Kittler, J. “A novel data association algorithm for object tracking in clutter with application to tennis video analysis,” *In Computer Vision and Pattern Recognition*, 2006 IEEE Computer Society Conference on (Vol. 1, pp. 634-641). IEEE.

[10] Chen L, Pan Y, Chen Y.“ A Study of Shuttlecocks Trajectory in Badminton,” *Journal of sports science medicine*, 2009, 8(4): 657-662.

[11] Zhang Z. “A flexible new technique for camera calibration,” *Pattern Analysis and Machine Intelligence*, IEEE Transactions on, 2000, 22(11): 1330-1334.

[12] Torr P H S, Zisserman A. MLESAC: “A new robust estimator with application to estimating image geometry,” *Computer Vision and Image Understanding*, 2000, 78(1): 138-156.

[13] Ashutosh Saxena, Min Sun, Andrew Y. Ng, “Make3D: Learning 3D Scene Structure from a Single Still Image,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.31, no. 5, pp. 824-840, May 2009, doi:10.1109/TPAMI.2008.132

[14] Smeulders, Arnold WM, et al. “Visual tracking: an experimental survey,” *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 36.7 (2014): 1442-1468

[15] Wang, X., Ablavsky, V., Shitrit, H. B.,Fua, P. “Take your eyes off the ball: Improving ball-tracking by focusing on team play,” *Computer Vision and Image Understanding* 119 (2014): 102-115.

[16] Yan, Fei, William Christmas, and Josef Kittler. ”Ball Tracking for Tennis Video Annotation.“ Computer Vision in Sports. Springer International Publishing, 2014. 25-45.