EXTRACTION OF BASEBALL TRAJECTORY AND PHYSICS-BASED VALIDATION FOR SINGLE-VIEW BASEBALL VIDEO SEQUENCES

Wei-Ta Chu[1], Chia-Wei Wang[1], and Ja-Ling Wu[1,2]
[1]Department of Computer Science & Information Engineering,
[2]Graduate Institute of Networking and Multimedia,
National Taiwan University, Taiwan
{wtchu,nacci,wjl}@cmlab.csie.ntu.edu.tw

ABSTRACT

To enrich the viewing experience of baseball games and provide some clues for enhancing pitcher's performance, we propose a Kalman filter-based approach to track ball trajectory from single-view pitching sequences. Without setting extraordinary equipments in stadiums or other sensing instruments, this approach robustly extracts ball trajectory for pitching sequences captured from TV channels or downloaded from the Internet. To validate the detected ball trajectories, we investigate the characteristics of ball trajectories on the basis of a baseball physical model. The effectiveness of ball trajectory extraction and ball position detection are presented.

1. INTRODUCTION

Many techniques based on color, motion, caption [1], and audio information [2] have been proposed for sports video analysis. Most approaches focus on detecting predefined events in games and then generate game summarization or abstraction. On the other hand, some implicit game statistics, which may be helpful in tactic analysis or improving athlete's performance, have drawn only little attention. Examples of this information include possession time in soccer games and various pitches in baseball games.

Recently, approaches based on ball trajectory have been proposed to facilitate such kind of implicit game status extraction. Yu et al. [3] detect and track ball trajectory in soccer games to perform possession and play-break structure analysis. For baseball games, the main scope of this paper, the well-known K Zone system [4] is to track pitched ball trajectory. Two cameras (locating high above home plate and the first base) and three subsystems are equipped to achieve real-time ball tracking in broadcasting baseball games. More specifically, Theobalt [5] et al. track position, velocity, rotation axis, and spin of the pitched ball along its trajectory with low-cost commodity.

These approaches are only applied to some specific games or should be equipped with special tracking instruments. Nowadays, tremendous pitching video sequences can be accessed on the Internet [6], while such entertaining functionality is not provided. Techniques that automatically extract ball trajectory without specific equipments are therefore worth developing to enrich the experience of watching baseball games. In this paper, we focus on extracting ball trajectory from single-view pitching video sequences. We apply a Kalman filter-based approach [7] to perform ball tracking and extract trajectories. This approach robustly provides good tracking performance even some real ball candidates are missing in some frames. The detected trajectory is further validated by the baseball physical model given in [8].

The rest of this paper is organized as follows. Section 2 gives the system overview. Section 3 presents the proposed ball detection process. Section 4 addresses the trajectory process and the physics-based trajectory validation. Performance evaluation is given in Section 5, and Section 6 concludes this paper.

2. SYSTEM OVERVIEW

Figure 1 illustrates the proposed baseball trajectory detection process. Given a pitching video sequence, ball candidates in each frame are first detected by checking color, position, size, and shape information. Several ball candidates may be extracted from one frame, while the real ball object may be misdetected because the ball is occluded by players or is merged into white regions. Therefore, the trajectory process is applied to track the ball in video frames and generate trajectory candidates. This process deals with ball tracking, trajectory interpolation and extension. For a pitching sequence, only one trajectory is valid after checking physical characteristics of the detected trajectory candidates. The planar ball position of each frame can then be determined. The proposed approach works well for different types of pitching conditions (games in MLB vs. CPBL [9] and righty vs. lefty).

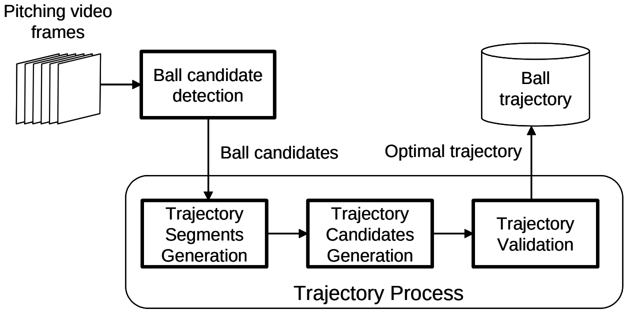

Figure 1. The system flowchart.

3. BALL CANDIDATE DETECTION

Given a pitching video sequences, we first detect white background regions that always remain white in the whole sequence (usually 10~18 frames in 30-fps video sequences). For each video frame, the pixels in white background regions are neglected, and white objects are viewed as ball candidates if they meet several constraints, including color, position, size, and shape. Figure 2 illustrates the process of ball candidate detection.

1) Color filter: The color of a baseball is similar to white even in different broadcasting situations. Therefore, the objects whose color is not close to white are filtered out. The object is close to white if all its color components in RGB channels are larger than 150.
2) Position filter: The ball always flies within a specific region, either in different broadcasting styles, right-hander or left-hander. We discard all objects higher than (1/5)*(frame height) and lower than (5/4)*(frame height).
3) Size filter: Although the size of a baseball may be different in various game broadcasts, it falls within a specific range. This filter sieves out pixel-size white noises or massive objects caused by the player's white uniform or advertisement boards. In our experiments, the resolution of video frames is 352 $\times$ 240 pixels, and reasonable planar ball size ranges from 2 to 10 pixels.
4) Shape filter: Although the high-speed flying ball is not a perfect circle on screen, it would not deform too drastically. Therefore, objects that are far from circle are filtered out. An object's radius *r* is defined as the greater value among its width and height, and it is viewed as a circle if the ratio of object area to πr^2 is larger than 0.3.

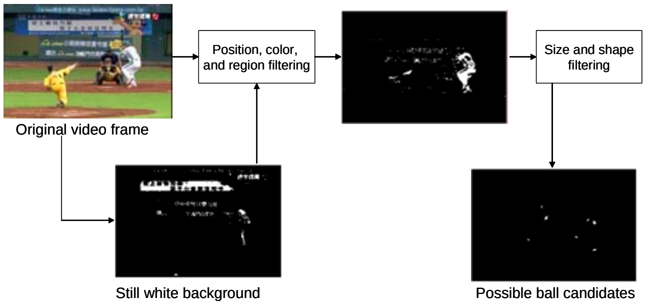

Figure 2. The flowchart for ball candidate detection.

After applying these filtering processes, reasonable ball-like objects are detected. However, many of them are noises or none of them is the real ball. The trajectory process described in the next section tracks these ball candidates and finds the optimal trajectory.

4. TRAJECTORY PROCESS

The trajectory process consists of three steps. We first connect neighboring ball candidates in adjacent frames to form trajectory segments. If the real ball is completely detected in the whole pitching sequences, we can feel free to say that one of these trajectory segments is the true ball trajectory. However, ball is often misdetected because of occlusion, merging, or deformation, and the trajectory is cut into short segments. Therefore, the process of trajectory candidates generation tackles with constructing complete trajectories. Furthermore, an elaborate trajectory validation process is developed to filter out abnormal (in terms of baseball physics) trajectories derived from detection noises.

4.1. Trajectory Segments Generation

The main challenge of trajectory generation is that the ball often overlaps with white objects and is not detected (filtered out) in the process of ball candidate generation. Therefore, the developed trajectory process should estimate the missing ball positions and generates reasonable trajectory candidates. In this work, we apply a Kalman filter-based approach to track the ball positions.

In general, the Kalman filter describes a system as:

$$\mathbf{x}_{\mathrm{k}} = \mathbf{A}_{\mathrm{k}}\mathbf{x}_{\mathrm{k-1}} + \mathbf{w}_{\mathrm{k}}, \tag{1}$$

$$\mathbf{z}_{\mathrm{k}} = \mathbf{H}_{\mathrm{k}}\mathbf{x}_{\mathrm{k}} + \mathbf{v}_{\mathrm{k}}, \tag{2}$$

where $\mathbf{x}_{\mathrm{k}}$ is the state vector (representing the vector of estimated ball position at the *k*th frame), $\mathbf{A}_{\mathrm{k}}$ is the system evolution matrix, and $\mathbf{w}_{\mathrm{k}}$ is the system noise vector. $\mathbf{z}_{\mathrm{k}}$ is the vector of measurements (positions of ball candidates), $\mathbf{H}_{\mathrm{k}}$ is the unit array, and $\mathbf{v}_{\mathrm{k}}$ is the measure noise vector.

We first find a trajectory seed to start the process of Kalman filtering. A trajectory seed is a pair of ball candidates in two adjacent frames and the Euclidean distance between them is less than a given threshold. In this work, two ball candidates are viewed as a seed if both their vertical and horizontal distances are less than 15 pixels.

After using the found seed to estimate the system evolution matrix $\mathbf{A}_{\mathrm{k}}$, we grow the trajectory forward along the time axis. Suppose that an existing trajectory ends at the *i*th frame, the ball position in the (*i+1*)-th frame is estimated by the Kalman filter. If there are ball candidates close to the estimated position, the trajectory grows and we update the parameters of Kalman filter. The classical predictor-corrector process [7] repeats until all video frames are analyzed or no close candidates can be found as the basis for trajectory growing.

Figure 3 shows an example of the detected trajectory segments. The distance at y-axis denotes the Euclidean distance between ball candidates and the left-top corner of the frame. Note that the true ball trajectory may not exist at all frames. Thus, we have to concatenate these trajectory segments in a reasonable way and construct trajectory candidates which last for the whole pitching duration.

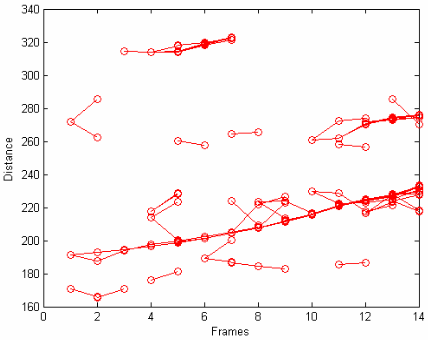

Figure 3. An example of detected trajectory segments.

4.2. Trajectory Candidates Generation

To generate trajectory candidates from the found trajectory segments, a process that consists of three stages is developed.

1) Find stage: For each trajectory segment T_i, which ends at the e_i-th frame, find the trajectory segments T_j, which starts before the (e_i+5)-th frame (assume it starts at the s_j-th frame) and the distance between their end points at the e_i-th frame and the s_j-th frame is less than a threshold.
2) Connect stage: If T_i is longer, use a polynomial to fit T_i and estimate its ball position at the s_j-th frame. If the distance between the estimated ball position and T_j's ball candidates is

less than a threshold, connect them. Otherwise, ignore this segment pair. Similar process is performed if T_j is longer.

3) Extend stage: If the connected trajectory segments are longer than (L-4) frames and shorter than L frames, where L is the number of total frames of a pitch, a polynomial is used to fit this trajectory and estimate the rest of this trajectory.

The trajectory generation process may generate many trajectory candidates that are drastically different. However, only one of them is the true ball trajectory. The following section describes how we find the optimal trajectory from these trajectory candidates.

4.3. Physics-based Trajectory Validation

4.3.1. Physical Model of Ball Trajectory

There have been many literatures on aerodynamics of baseball, by which we can confirm the reasonability of the generated trajectory candidates. According to the physics of baseball [8], the trajectory of a baseball can be roughly determined by velocity, rotation axis, and spin rate. By adjusting these parameters, we simulate trajectories of a pitching baseball and characterize its trajectory vectors. The positions of a baseball in x (right of batter), y (up from batter), and z (towards batter) directions can be formulated as:

$$
\begin{aligned}
x_{t+1} &= x_t + v_x(t)\times t + \frac{1}{2}a_x(t)\times t^2,\\
y_{t+1} &= y_t + v_y(t)\times t + \frac{1}{2}a_y(t)\times t^2,\\
z_{t+1} &= z_t + v_z(t)\times t + \frac{1}{2}a_z(t)\times t^2,
\end{aligned}
\qquad (3)
$$

where x_t is the horizontal position at time t, $v_x(t)$ is the velocity component in horizontal direction, and $a_x(t)$ is the corresponding acceleration. Related to the releasing angle and beginning velocity v_0, the x, y, z components of velocity are:

$$
\begin{aligned}
v_x(0) &= v_0 \times \sin(aim_x \times \frac{\pi}{180}),\\
v_y(0) &= v_0 \times \sin(aim_y \times \frac{\pi}{180}),\\
v_z(0) &= v_0 \times \cos(aim_x \times \frac{\pi}{180}) \times \cos(aim_y \times \frac{\pi}{180}),
\end{aligned}
\qquad (4)
$$

where aim_x (aim_y) is the included angle between z-axis and the projection of releasing ball on xz (yz) plane, as shown in Figure 4.

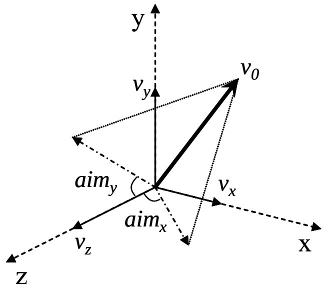

Figure 4. Velocity components of the releasing baseball.

The corresponding evolution of velocity becomes:

$$
\begin{aligned}
v_x(t+1) &= v_x(t) + a_x(t)\times t,\\
v_y(t+1) &= v_y(t) + a_y(t)\times t,\\
v_z(t+1) &= v_z(t) + a_z(t)\times t,
\end{aligned}
\qquad (5)
$$

and the corresponding evolution of acceleration is:

$$
\begin{aligned}
a_x(t) &= B \times s_y \times v_z(t) - fv \times v_0 \times v_x(t),\\
a_y(t) &= B \times s_x \times v_z(t) - G - fv \times v_0 \times v_y(t),\\
a_z(t) &= B \times \left(s_y \times v_x(t) - s_x \times v_y(t)\right) - fv \times v_0 \times v_z(t),
\end{aligned}
\qquad (6)
$$

where B and fv are coefficients of spin and air friction, G is the acceleration of gravity, s_x is the spin rate with rotation axis x. In each direction, the evolution of acceleration is affected by the force evoked by spin and the drag force. With these formulas, given the beginning velocity v_0, spin rate in x and y directions (s_x and s_y), we can simulate the ball trajectories in different conditions.

To find the criterion of valid ball trajectories, we simulate trajectories with different parameter sets, which include almost all possible parameters a pitcher can evoke. Table 1 shows the ranges of simulation parameters. Note that in physics books the orientation of rotation axis should also be considered. However, in our experiments, the difference of rotation orientation affects slightly and can be neglected.

Table 1. Ranges of simulation parameters

Parameters	Range
v_0	60, 70, 80, 90, 100 (mph)
s_x and s_y	-600, -500, …, 400, 500, 600 (rad/s)
aim_x and aim_y	-5°, -4°, …, 3°, 4°, 5°

For each simulated trajectory, we compute the included angle of two adjacent flying vectors:

$$
\theta_i = \cos^{-1}\left(\frac{\mathbf{v}_i \bullet \mathbf{v}_{i-1}}{\left\|\mathbf{v}_i\right\| \times \left\|\mathbf{v}_{i-1}\right\|}\right), \qquad (7)
$$

where $\mathbf{v}_i = \left(x_i - x_{i-1}, y_i - y_{i-1}, z_i - z_{i-1}\right)$.

After simulating all possible trajectories, the included angles in various conditions are gathered to be the reference for trajectory validation. Note that we only simulate overhand pitches, while the statistics of underhand pitches may not vary drastically. Figure 5 shows the angle histogram of two adjacent 3-dimensional (3-D) vectors. In this histogram, we can see that all legal included angles fall into the range less than 1.2°. This characteristic forms the limitation of a legal trajectory.

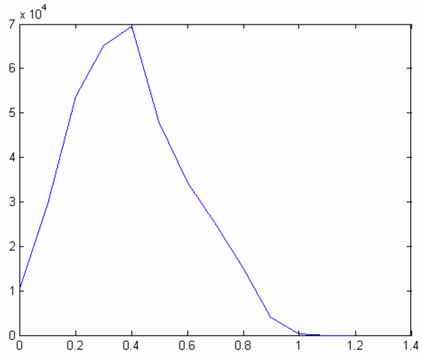

Figure 5. The angle histogram of trajectory vectors.

4.3.2. Trajectory Validation via Physical Limitation

With the aid of physical limitation derived from trajectory simulation, we can filter out abnormal trajectory candidates. However, what we extract from single-view video sequences are 2-

D ball trajectories, in terms of pixels. We should estimate the depths (ball positions at the z-axis) so that the constraints described above can be applied. In this work, we estimate the proportion of vertical movement (movement at the y-axis) to depth in the simulation process. After trajectory simulation, it is estimated as 0.0558. This ratio matches the naïve estimation in the real world. If the height of a pitcher plus mound is roughly 2 meters, the average vertical movement of various pitches is about 1 meter. This ratio can be calculated by dividing average vertical movement with the distance from the mound to the home plate (18.44 meters). On the other hand, the average vertical movement in our dataset is estimated as 38.1736 pixels. Proportionally, the depth of the detected 2-D trajectories is estimated as 38.1736/0.0558 ≈ 684 pixels. The depth of each ball candidate is then obtained from the estimated depth divided by the frame number of this sequence. By this way, we not only extract planar trajectory from single-view sequences, but also estimate its 3D model automatically.

For each detected 3-D trajectory candidate, the included angle between two adjacent vectors is computed. A trajectory candidate is viewed as abnormal if one of the included angles of its vectors is larger than 5°. This threshold is set according to the limitation derived from Figure 5, and is loosed to endure noises caused by detection or tacking errors.

5. EVALUATION

We use 38 pitching sequences, including fastball, splitter, slider, and curveball, for evaluation. The ground truth of ball position (center of the ball) is defined manually. The distance, in terms of pixels, between ground truth and the estimated ball positions is calculated. Table 2 shows the average and maximum pixel differences in four types of pitching. Note that we use 352×240 pixels MPEG-1 bitstreams, and the size of planar ball usually ranges from 4×4 to 6×6 pixels. We can see very good ball detection accuracy in Table 2. The reported differences are relatively small to ball size and can be viewed noises derived from defining ground truth manually. Moreover, assume that the error from human-defining ground truth is consistent in different types of pitches, the values in Table 2 also show some interesting results. Fastball goes straight and is relatively easy to be estimated, while curveball turns drastically and raises the difficulty of accurate tracking and detection. Splitter and slider curve between fastball and curveball and show medium differences.

Table 2. Detection performance in terms of pixel differences

Pitch type	Avg. pixel diff.	Max. pixel diff.
Fastball (18 seq.)	0.79	1.73
Splitter (5 seq.)	1.02	1.91
Slider (9 seq.)	1.28	2.48
Curveball (6 seq.)	1.28	2.71

Figure 6 shows a detection result that juxtaposes the real ball trajectory and the detected one. More results can be seen at http://www.cmlab.csie.ntu.edu.tw/~wtchu/baseball/trajectory.html.

6. CONCLUSION

We develop a system to automatically extract baseball trajectory from single-view pitch video sequences. By checking color, position, size, and shape information, ball candidates in each video frame can be detected. A Kalman filter-based approach is applied to track the ball position and generate trajectory segments. On the basis of trajectory segments, a process is designed to generate trajectory candidates, which last for the whole pitching duration. We evaluate the reasonability of each trajectory and select the best one to obtain the final trajectory result based on a baseball physical model. The experimental results show the effectiveness of this approach. In the future, the results of ball trajectory will be applied to either entertainment or game analysis.

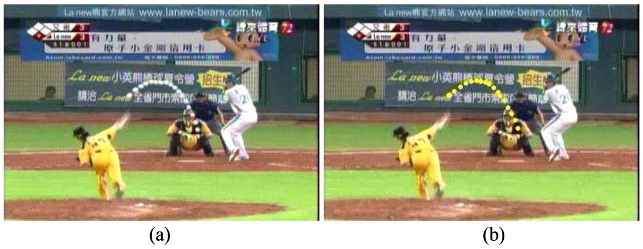
(a) (b)

Figure 6. Comparison of (a) the true ball trajectory and (b) the extracted trajectory.

7. ACKNOWLEDGEMENT

This work was partially supported by the National Science Council and the Ministry of Education of ROC under the contract No. NSC94-2752-E-002-006-PAE, NSC94-2622-E-002-024, and NSC94-2213-E-002-078.

8. REFERENCES

[1] D. Zhang and S.-F. Chang, "Event detection in baseball video using superimposed caption recognition," *Proc. of ACM Multimedia Conference*, pp. 315–318, 2002.

[2] Y. Rui, A. Gupta, and A. Acero, "Automatically extracting highlights for TV baseball programs," *Proc. of ACM Multimedia Conference*, pp. 105–115, 2000.

[3] X. Yu, C. Xu, H.W. Leong, Q. Tian, Q. Tang, and K.W. Wan, "Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video," *Proc. of ACM Multimedia Conference*, pp. 11-20, 2003.

[4] A. Guezic "Tracking pitches for broadcast television," *IEEE Computer*, Vol. 35, No. 3, pp. 38-43, 2002.

[5] C. Theobalt, I. Albrecht, J. Haber, M. Magnor, and H.-P. Seidel, "Pitching a baseball: tracking high-speed motion with multi-exposure images," *Proc. of ACM SIGGRAPH*, pp. 540-547, 2004.

[6] Major League Baseball (MLB), http://www.mlb.com

[7] G. Welch and G. Bishop, "An introduction to the Kalman filter," Technical Report no. TR 95-041, University of North Carolina at Capel Hill, 2004.

[8] R.K. Adair, "The Physics of Baseball," Harper Collins, New York, 2002.

[9] Chinese Professional Baseball League (CPBL), http://www.cpbl.com