A GENERAL FRAMEWORK FOR 3D SOCCER BALL ESTIMATION AND TRACKING

Jinchang Ren, James Orwell, Graeme A. Jones, Ming Xu

Digital Imaging Research Centre, Kingston University
Kingston upon Thames, KT1 2EE, UK.
{j.ren,j.orwell, g.jones, m.xu}@kingston.ac.uk

ABSTRACT

A general framework for automatic 3D soccer ball estimation and tracking from multiple image sequences is proposed. Firstly, the ball trajectory is modelled as planar-curves in consecutive *virtual vertical planes*. These planes are then determined by two ball positions with accurately estimated height, namely *critical points*, which are extracted by curvature thresholding and nearest distance of 3D lines from single or multiple views respectively. Finally, unreliable or missing ball observations are recovered using geometric constraints and polynomial interpolation. Experiments on video sequences from different cameras, with over 5000 frames each, have demonstrated a comprehensive solution for accurate and robust 3D ball estimation and tracking, with over 90% ball estimations within 2.5 metres of manually derived ground truth.

1. INTRODUCTION

With the development of computer vision and multimedia technologies, sports video analysis has increasingly important applications in content-based video indexing, retrieval and visualization [1,2,3]. Through motion analysis and tracking of the players and the ball, additional information can be extracted for better comprehension of video contents and sports events, including video content annotation, summarization, team strategy analysis and verification of referee decisions, as well as further 2D or 3D reconstruction and visualization [3-8].

Accurate tracking of players and the ball is essential for soccer video analysis and reconstruction. Although players can be successfully detected and tracked based on colour and shape [1,4,6], similar methods cannot be extended to ball detection and tracking for several reasons: first, the ball is too small, exhibits irregular shape, variable size and unstable colour when moving fast (see Figure 1). Second, the ball is frequently occluded by players or flying out of view from many of the cameras. Finally, the ball is mostly travelling above the ground, which necessitates 3D tracking for accurate positioning. Therefore, 3D ball position estimation and tracking is, arguably, the most important challenge in soccer video reconstruction.

Generally, multiple cameras are applied to estimate and track 3D balls in two steps. Firstly, the ball is detected and tracked in single views independently. Then, 2D ball positions from different camera views are integrated to attain 3D positions using known motion models [4-6]. In Ohno *et al* [4], 3D ball trajectory is modelled by considering air friction and gravity but depends on unsolved initial 3D velocity. In Matsumoto *et al* [5], 2D ball is detected by template matching and tracked by epipolar line constraints between multiple cameras. Bebie, and H. Bieri [6], model 3D trajectory segments by Hermite spline curves. However, about one-fifth of the ball positions need to be set manually before estimation. In Kim *et al* [7] and Reid and North [8], reference players and shadows are utilized in the estimation of 3D ball positions. These are unlikely to be robust as the shadow positions depend more on light positions than camera projections.

Figure 1. Ball samples in various size, shape and colours.

In this paper, a general framework for a complete solution is proposed for 3D ball estimation and tracking from real soccer videos. Firstly, 3D ball movements are modelled as planar-curves in a series of vertical planes. Then, at least two critical points with reliable height estimate are determined in every curve. If the ball is only detected from a single view, the critical points are located as the bouncing points as extrema of curvature. Otherwise, the 3D critical points can be positioned as the centroid of intersection points of each pair of 3D lines from different views, where the 3D lines are defined by ground positions of detected balls and their corresponding camera positions. Based on each of the two critical points, a *virtual vertical plane* (VVP) can be determined. Then, all the unreliable ball observations between these two critical positions are determined by triangular constraints, as the intersection points between the VVP and lines from the projected ball positions. Finally, ball positions in frames where no observations available are estimated by polynomial interpolation to generate a continuous 3D ball trajectory in all frames. The accuracy of the system has been validated us-

0-7803-8554-3/04/$20.00 ©2004 IEEE.

ing several minutes of ground truth data manually recovered from the image sequences.

2. SEGMENT BASED BALL MOTION AND TRAJECTORY MODELLING

2.1. Segment-based ball trajectory modelling

During a soccer game, the ball is moving regularly from one place to another. Its directions will change suddenly if and only if it touches the ground, a player, or goal post, etc. We assume that a ball's trajectory will form a curve in a virtual plane. In a special case, the curve will become a line when the ball is rolling on the ground. Therefore, we can model the ball movement as separate segments in consecutive virtual vertical planes. In each vertical plane π, the ball will generate a single trajectory curve. Then, the complete ball trajectory can be modelled as an adjacent sequence of planar curve segments.

In order to obtain the complete ball trajectory, we should first determine each of its trajectory segments in the vertical plane π. To determine a virtual vertical plane, at least two known ball positions (critical points) on the vertical plane, are required. The process for automatic locating of critical points and the virtual vertical plane will be discussed in section 3.

2.2. Internal ball estimation within each segment

When the location of a virtual plane π is determined, the 3D positions of detected ball observations can be recovered by triangulation between projected 2D camera observations of the ball and plane π as shown in Figure 2.

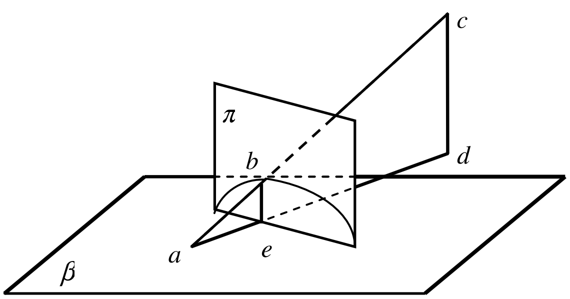

Figure 2. Geometric relationship among camera and ball with vertical and ground planes.

Let us define $X_p = (x_p, y_p, z_p)$ as the world coordinates of any point p. In Figure 2, X_e is the intersection with the ground plane of the projection of a ball observation detected in camera c located at X_c, and X_b is the required 3D position of the ball on the vertical plane. Therefore, X_b is given from the intersection point of plane π and the line from X_c to X_a. Points X_d and X_e are the normal projections of c and b respectively onto the ground plane. The coordinates of points X_a, X_b, X_c, X_d and X_e in Figure 2 satisfy:

$$z_a = 0 \tag{1}$$

$$(x_d, y_d, z_d) = (x_c, y_c, 0) \tag{2}$$

$$(x_e, y_e, z_e) = (x_b, y_b, 0) \tag{3}$$

With known plane π and points a and d, we can recover x_b and y_b from e as the intersection point of π and the line from a to d. As the two triangles Δacd and Δabe are similar, we also have:

$$\frac{\|X_b - X_e\|}{\|X_c - X_d\|} = \frac{\|X_a - X_e\|}{\|X_a - X_d\|} \tag{4}$$

$\|X_b - X_e\| = z_b$ and $|cd| = z_c$, we obtain z_b as:

$$z_b = \frac{\|X_a - X_e\|}{\|X_a - X_d\|} \cdot z_c \tag{5}$$

Thus the 3D ball position X_b has been recovered.

2.3. Undetected ball positions estimation

For those frames without detected balls, the ball positions are decided by polynomial interpolation in a curve on the corresponding vertical planes. In addition to two extracted critical points, with the third estimated ball position, a parabola trajectory curve can be attained. If we have more additional ball positions, more accurate trajectories can be estimated. In the extreme case, in which there is no other ball detected except for the two critical points, the interpolated ball trajectory degrades to a line.

3. CRITICAL POINTS AND VIRTUAL VERTICAL PLANE DETERMINATION

To define a vertical plane, at least two critical points are necessary. Critical points are those ball positions, which can be determined accurately from the ball observations and their corresponding camera positions. In the following section we will give the process for automatic determination of critical points as well as the vertical plane.

3.1. Critical point identification in a single view

From a single camera view, ground plane ball positions can firstly be obtained as $a(n) = (x_n, y_n)$. Define $\mathbf{x}(p)$ represents the coordinate vector of point p as $[x_p, y_p]^T$ or $[x_p, y_p, z_p]^T$ in 2D and 3D respectively. Then, the velocity $\mathbf{v}(n) = [v_x, v_y]^T$ is defined as:

$$\mathbf{v}(n) = \mathbf{x}(a(n+1)) - \mathbf{x}(a(n)) \quad 6)$$

Ball position $a(n)$ is a critical point if the ball changes its moving direction at frame n and satisfies:

$$\cos^{-1} \frac{\mathbf{V(n-1)} \cdot \mathbf{V(n)}}{\| \mathbf{V(n-1)} \| \cdot \| \mathbf{V(n)} \|} > \theta_0 \quad (7)$$

where θ_0 is a given threshold.

Then, the height of the ball is estimated as zero if there are no players or other objects near the critical point. Otherwise, it is decided by the relative position of the critical point and the object it touched.

3.2. Critical points detection from multiple views

Assume a 3D ball b is observed from two cameras c_1 and c_2 with projected positions b_1 and b_2 on the ground plane β (see Figure 3). We will estimate b from b_1, b_2, c_1 and c_2.

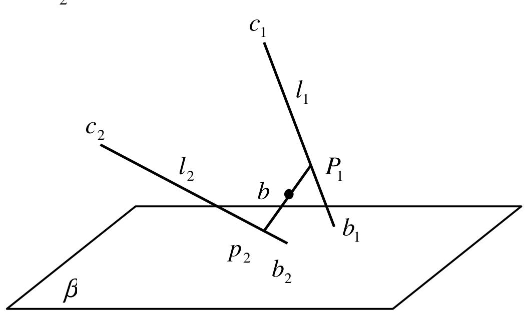

Figure 3. 3D ball b estimation from cameras c_1 and c_2 with projected ball positions b_1 and b_2.

Let l_1 and l_2 be two lines from c_1 to b_1 and c_2 to b_2 respectively, thus their intersection point should be the optimal estimation of b. Unfortunately, l_1 and l_2 usually have no intersection point due to errors caused by camera calibration. Therefore, we define b as the point which has minimum distance to both of these two lines.

We define two points p_1 and p_2 on lines l_1 and l_2 respectively. We require the line from p_1 to p_2 be a common perpendicular of l_1 and l_2. Then b should be on the line between p_1 and p_2. The points p_1 and p_2 can be determined by:

$$f_1[\mathbf{x}(p_1)] = 0 \quad (8)$$

$$f_2[\mathbf{x}(p_2)] = 0 \quad (9)$$

$$[\mathbf{x}(p_1) - \mathbf{x}(p_2)] \cdot [\mathbf{x}(c_1) - \mathbf{x}(b_1)] = 0 \quad (10)$$

$$[\mathbf{x}(p_1) - \mathbf{x}(p_2)] \cdot [\mathbf{x}(c_2) - \mathbf{x}(b_2)] = 0 \quad (11)$$

In equations (8) and (9), the equations of the lines $f_1[\cdot]$ and $f_2[\cdot]$ constraint p_1 and p_2 to the lines l_1 and l_2 respectively. Equations (10) and (11) define the line between p_1 and p_2 to be perpendicular to l_1 and l_2.

Points p_1 and p_2 can be estimated accurately from lines l_1 and l_2, which are defined by known points c_1, b_1, c_2 and b_2. Assuming different cameras have similar calibration error, then the 3D ball position b is estimated as the middle point of p_1 and p_2.

If the ball is observed in more than 2 cameras, we will first find the estimated 3D ball position of each pair of different views, and the final ball position b is estimated as the average of these estimated points.

3.3. Virtual vertical plane determination

Let r and s be two known critical points in a seg-curve on a virtual plane π, the plane π can be easily determined as follows: Firstly, locate points r' and s' on the ground plane β with $rr' \perp \beta$ and $ss' \perp \beta$, then we have a line $r's'$ on β. Finally, π is determined as a plane through $r's'$ and perpendicular to β.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

In our system, the soccer videos were captured by 8 fixed cameras positioned around a football stadium. All the cameras were manually calibrated before tracking and 3D positioning. All the moving objects are detected and tracked in every image sequence with ball candidates being filtered on normalized size, shape and colour features. Next, 3D positions of these detected balls are estimated and tracked in world coordinates based on our model. Finally, the 3D ball trajectory is visualized along with tracked players on a virtual playfield.

Figure 4 shows a plane view of the estimated 3D ball positions when a ball was kicked out by the goalkeeper. Estimated ball positions are shown as a magenta trajectory. The grey trajectories are ground plane projections of ball positions from different camera views. The brown line in front of the 3D ball trajectory refers to path of ground truth, along which the actual ball trajectory should follow. Player positions are marked by black or white circles with tails representing recent trajectory. For comparison, two single-view ball trajectories are also given in image planes and projected ground plane, respectively.

To evaluate the accuracy of our 3D ball tracking model, manually derived ground truth (GT) was extracted from about every 25th frame in the image sequences, providing 239 GT positions in 8 sequences. For the frames between two GT frames, an estimated GT position is obtained from linear interpolation. The distance between positions of estimated actual ball and ground truth positions are obtained.

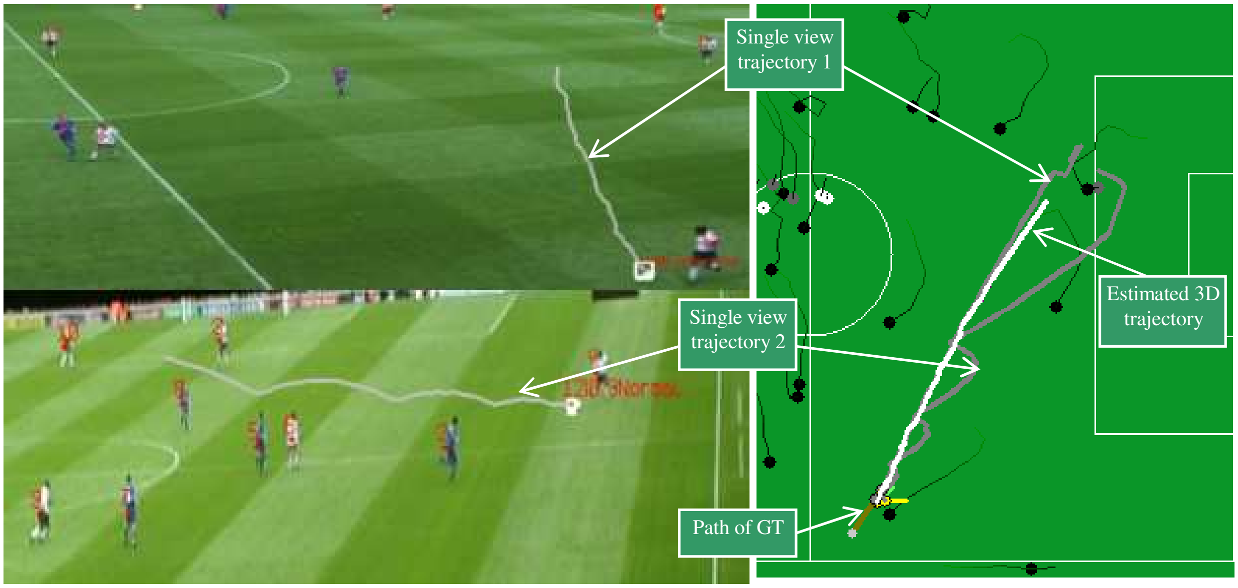

Figure 4. Estimated 3D ball trajectory with players compared with two single-view trajectories and ground truth.

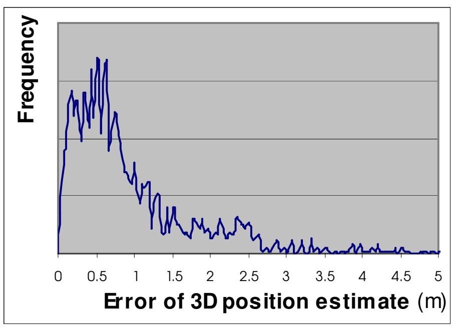

Figure 5. Tracking accuracy compared with ground truth.

Figure 5 plots the histogram showing distance between the automatic estimate and the manually recorded ground truth. The integration of the histogram represents the probability of estimations within given distance of the ground truth, thus we have more than 90% of ball positions accurate to within 2.5 meters. The maximum discrepancy between the ground planes of each calibrated camera is of the same order of magnitude, which puts this result in context. Currently, positions can be estimated in about 56% of frames as the ball is frequently occluded, camouflaged by the crowd, or out of play.

5. CONCLUSIONS

Using geometric reconstruction techniques, we propose a general framework for 3D soccer ball estimation and tracking from multiple image sequences. The experimental results from long sequences have shown reasonably accurate performance of our model. Future investigations will focus on reasoning in occluded ball situations, handling out of game situations, and relaxing the assumptions of vertical planes to handle more complex ball trajectories and considering the effects of air friction and ball spin.

6. ACKNOWLEDGEMENT

This work forms part of the INMOVE project, supported by the European Commission IST 2001-37422.

7. REFERENCES

[1] Y. Gong, T. S. Lim, H. C. Chua, H. J. Zhang, and M. Sakauchi, "Automatic Parsing of TV Soccer Programs", in Proc. Multimedia Comp. and Systems, 1995, pp. 167-174.

[2] D. Yow, B. L. Yeo, M. Yeung, and B. Liu, "Analysis and Presentation of Soccer Highlights from Digital Video", in Proc. ACCV, 1995, pp. 499-503.

[3] G. Pingali, A. Opalach, and Y. Jean, "Ball Tracking and Virtual Replays for Innovative Tennis Broadcasts", in Proc. ICPR, 2000, pp. 152-156

[4] Y. Ohno, J. Miura, and Y. Shirai, "Tracking Players and Estimation of the 3D Position of a Ball in Soccer Games", in Proc. ICPR, 2000, pp. 145-148.

[5] K. Matsumoto, S. Sudo, H. Saito, and S. Ozawa, "Optimized Camera Viewpoint Determination System for Soccer Game Broadcasting", IAPR Workshop MVA, 2000, pp. 115-118.

[6] T. Bebie, and H. Bieri, "SoccerMan – Reconstructing Soccer Game from Video Sequence", in ICIP, 1998, pp. 898-902.

[7] T. Kim, Y. Seo, and K. S. Hong, "Physics-based 3D Position Analysis of a Soccer Ball from Monocular Image Sequences", in Proc. ICCV, 1998, pp. 721-726.

[8] I. Reid, and A. North, "3D Trajectories from a Single Viewpoint using Shadows", in Proc. BMVC, 1998, pp. 863-872.