BILLIARDS WIZARD:
A TUTORING SYSTEM FOR BROADCASTING NINE-BALL BILLIARDS VIDEOS

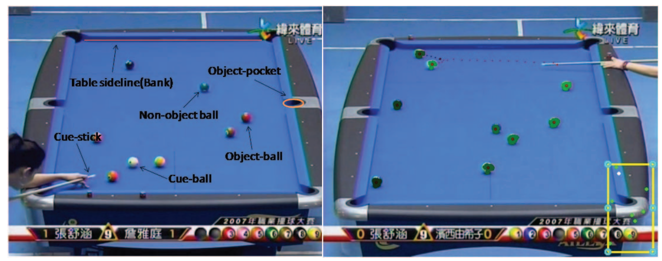
Chen-Wei Chou, Ming-Chun Tien, and Ja-Ling Wu

Graduate Institute of Networking and Multimedia
National Taiwan University

ABSTRACT

In this work, we propose a framework to build a billiards tutoring system based on broadcasting nine-ball video analysis. A robust table detection module is developed by mapping the displayed video frames to a predefined billiard table model. In addition, we detect balls and trace their positions at every time instant. The real-world spatial relationships between the table and the balls are used to provide the aiming and position play suggestions. Ball position information is also utilized to distinguish each play into corresponding event by a rule-based method. The experimental results are encouraging and are more comprehensive than existing works of billiards video analysis.

***Index Terms*—** Sports analysis, Billiards video analysis, Camera calibration



(a) (b)

Fig. 1. (a)The components in a general nine-ball billiards video frame.(b)A sample result of ball detection and tracking.

1. INTRODUCTION

Sports video analysis is becoming a flourish area in multimedia research. A large amount of studies for different kinds of sporting videos has been conducted on automatic event detection, structure analysis, summarization, highlight extraction, three-dimensional (3D) visualization, and so forth. Over the past decade, techniques proposed for analyzing soccer, baseball and tennis videos have been comprehensively studied. By contrast, there are relatively few works for the analysis of billiards videos. In a billiards game, the players usually spend a lot of time thinking and aiming at the balls, and the truly meaningful activities only occupy for a small fraction of time. From the viewpoint of audience, it would be helpful to have an effective tool to automatically detect the playing clips and identify special events from lots of plays.

Billiards is a very popular sport around the world with a long and rich history stretching from its inception in the 15th century. According to the type of ball colors and the rules in use, a billiards game can be classified into different categories, such as snooker, rotation, nine-ball, etc. A nine-ball game is a rotation game using the balls numbered from 1 to 9 and a cue-ball. A player is required to make the cue-ball first hit the object-ball, the ball with the lowest number on the table, before the cue-ball hit any other non-object ball or a table sideline. The game is won by legally pocketing the ball number 9. Fig.1(a) shows an example of a broadcasting nine-ball video frame, and points out ”cue-ball”, ”object-ball”, ”non-object ball”, ”cue-stick”, ”object-pocket”, and ”table sideline(bank)”.

The nine-ball game is known as a sport that needs a lot of skills, and people generally spending lots of money finding a professional tutor. In this work, we propose a billiards tutoring system for the beginners by providing professional recommendations, such as the aiming suggestion and the position play suggestion. The rest of this paper is organized as follows. We introduce the system framework in Section 2. Section 3 presents the video processing techniques for play segment extraction and table/ball detection. With the ball position information, a billiards tutoring system is conducted in Section 4. Section 5 shows the experimental results and Section 6 concludes our work.

2. SYSTEM FRAMEWORK

As illustrated in Fig.2, the proposed system framework is composed of two stages: the video processing stage and the tutoring stage. In the video processing stage, shot change detection is first adopted to segment videos into shots. Based on dominant color information and a robust table detection module, we classify shots into play shots or break shots. For each play shot, we further detect and track the balls along the time axis. In the tutoring stage, our system plays the role of either

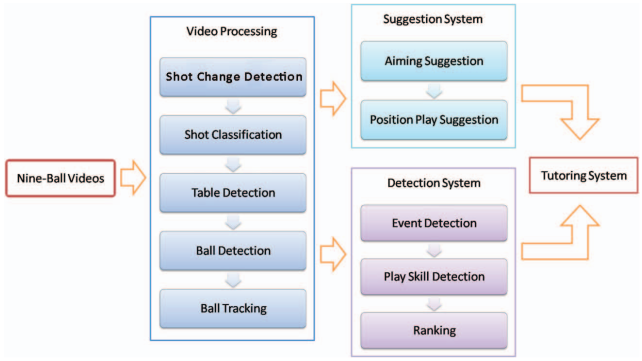


Fig. 2. System framework of the proposed Billiards Wizard.

a suggestion system or a detection system. In the suggestion mode, our system uses the ball detection results of the first frame of a play to give users the aiming suggestion, including the level of difficulty and a suggested aiming direction. Since most of nine-ball players are struggling in the position play, our system also provides the position play suggestion. In the detection mode, our system extract semantic events from all plays. The event detection results can be further applied to identify the skill difficulty, and a ranking mechanism is developed according to skill difficulty.

3. VIDEO PROCESSING

In billiards games, players and spectators are generally quiet, so we focus on video analysis rather than audio. Given a broadcasting billiards video, our system can automatically extract every play shot, which is the basis unit to conduct the table/ball position detection.

3.1. Shot Change Detection and Shot Classification

Billiards videos are composed of several play shots (shots in table view) and break shots (shots in non-table view). Since semantic events always occur in play shots, all play shots are extracted to do further analysis. We first apply a typical shot change detection method [1] based on histogram difference to segment videos into shots, and classify each shot into play or break according to the ratio of dominant color pixels. Frames in a play shot contain a large ratio of pixels possessing the table-color, which clearly would be the dominant color of these frames. However, the table color varies in different billiards matches or under different lighting conditions. Thus, we adaptively determine the color characteristics of the billiard table for each match on the basis of Gaussian mixture model [2]. In this work, we periodically adjust the dominant color ranges every ten minutes, according to the histograms collected from the latest five hundred frames. For each possible play shot, we detect the location of billiard table (see the details described in 3.2) and filter out some false alarms of play shot detection. If no table is detected in a play shot, the shot should be filtered out even if it contains a large amount of dominant color pixels.

3.2. Table Detection

The objective of table detection is to locate the table sidelines in video frames. Thus we can further investigate the real-world activities in the billiards videos. For all possible play shots, we detect the table sidelines based on the techniques of line detection and camera calibration [3]. We check all line candidates and identify the four border lines of the table as the ones enclosing the maximum non-dominant color area. With the calibration technique, we obtain the real-world locations of the table and the six pockets, and all position relations between these four border lines. If all position relations are correct, the true table sidelines are found; otherwise, we filter out this possible play shot.

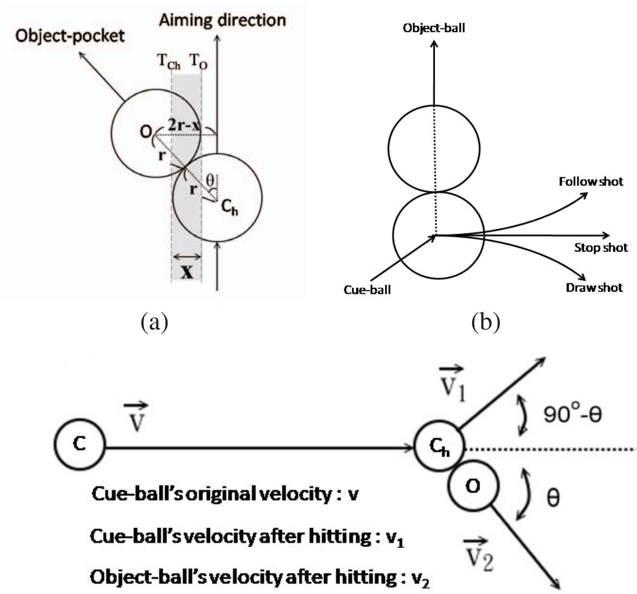
3.3. Ball Detection and Ball Tracking

To obtain reliable ball detection and tracking results, we utilize color, coordinate, and velocity information to cluster pixels in each frame of a play shot. With the results of shot classification and table detection, we take all non-dominant color pixels in the table region as foreground pixels. Since all balls on the table are included in foreground pixels, we cluster foreground pixels by K-means algorithm to obtain ball candidates. Each foreground pixel is described with a feature vector f containing its coordinates (x, y) , velocity (v_x, v_y), and the colors in HSI space (h, s, i) (i.e., $f = (x, y, v_x, v_y, h, s, i)$), where the velocity of each pixel is estimated by the Optical Flow algorithm [4]. We apply the Flood Fill algorithm[5] to the first frame of a play shot to determine the cluster number, and set the extracted group centers as the initial cluster centers of K-means clustering.

After clustering, each cluster corresponds to one ball on the table. To track the balls, we then define the dissimilarity of a cluster center p in the current frame and a cluster center p' in the previous frame as

$$D(p, p') = \sum_{i=-N}^{N} \sum_{j=-N}^{N} |I(x+i, y+j) - I'(x'+i, y'+j)|. \tag{1}$$

We set all the six object-pockets as leaving regions since a ball leaves the table legally when it is hit into any one of these pockets. A cluster disappearing in a leaving region will be removed, and the cluster number will then be decreased by one. On the other hand, if a cluster disappears in a non-leaving region, the cluster number and the corresponding cluster center will be the same as those in the previous frame. The later case occurs when balls are occluded by players or some other noises.



(a) (b)

(c)

Fig. 3. (a)The hitting moment of the cue-ball and an object-ball.(b)Three kinds of cue-ball directions after hitting.(c)The velocity and direction after hitting.

Fig.1(b) shows a sample result of ball detection and tracking. All foreground pixels are indicated by green pixels and the trajectory of each cluster is represented by red pixels. For each play shot, the ball moved first must be the cue-ball, and the ball hit by the cue ball is the object ball. Therefore, our system takes the first and the second moving cluster centers as the cue ball and the object-ball, respectively.

4. TUTORING SYSTEM

Based on the extracted ball trajectories, we develop a billiards tutoring system to help beginners who want to improve their billiards skills. When the user watches a play performed by a professional billiardist, the system will suggest which pocket is easier to pocket, how to goal, and how to make a good position play. The system can also automatically conduct event detection and hitting skill detection; therefore, users are able to know what kind of skills the player used in this play.

4.1. Aiming Suggestion and Position Play Suggestion

Even professional players cannot always hit the ball with accurate hitting angle. Hence we design this system to determine the aiming direction of the cue-ball for each pocket, and suggest the best one to the player. As illustrated in Fig.3(a), θ is the goal angle between the aiming direction and the object-pocket direction. Since an object-pocket is a little bigger than a ball in size, a range of angles is allowed to finish a pot. However, the difficulty of finishing a pot is not determined only by the range of angles, but also by the value of θ. For example, changing θ from $10°$ to $20°$ will lead to much more difference in the *object-ball path* than changing θ from $70°$ to $80°$.

By contrast, *hitting thickness* is a better index to represent the difficulty of a pot. We define the *hitting thickness* **x** as the distance between the two tangents T_{Ch} and T_O, where T_{Ch} and T_O denote the tangents to the cue-ball and the object-ball, respectively, and both T_{Ch} and T_O are parallel to the aiming direction of the cue-ball. In Section 3, we extract the real-world positions of the six pockets, the cue-ball, and the object-ball in the first frame of each play. And from Fig.3(a), we obtain the following equation

$$\sin\Theta = \frac{2r - x}{2r} \tag{2}$$

Therefore, the *hitting thickness* **x** can be obtained by

$$x = 2r(1 - \sin\Theta) \tag{3}$$

The larger the range of **x** is, the easier the pot is. And we suggest the easiest pocket to the user. For a billiards amateur, it is not easy to finish a pot by considering only the *hitting thickness*, so we show the aiming direction of the cue-ball as a reference for the player.

Since billiards rules allow the player to continue playing if he/she successfully finishes a pot, the position play which influence the next play is also very critical. For this reason, our system recommends users not only the best object-pocket but also how to perform a good position play, that is, how to locate the cue-ball after hitting. We first choose the easiest pocket for the next object-ball and determine the region of position play (***R***). Another problem is how to plan the cue-ball path so that the cue-ball can be located in ***R*** after pocketing the current object-ball. Fig.3(c) shows the cue-ball's path and velocity. According to the principle of conservation of momentum, we can approximately calculate the cue-ball direction and velocity after hitting. Eventually, we decide how many cushions in the cue-ball path and show the ideal cue-ball trajectory after hitting.

4.2. Event Detection, Hitting Skill Detection and Ranking

Each play shot comprises a special semantic event. We define six types of events as below, and apply a rule-based method to detect all the six events.

- Fault: the cue-ball is pocketed.
- Others: there is no ball pocked.
- Normal pot: the cue-ball pockets the object-ball and the *object-ball path* does not intersect any cushion.
- Combination shot: the cue-ball hits the object-ball, and the object-ball hits one or more non-object balls.
- Carom shot: the cue-ball pockets the object-ball, and the object-ball hits one or more non-object balls before pocketed.

Table 1. Performance of the billiards event detection.

Event Type	Precision	Recall
Combination shot	0.90 (18/20)	0.78 (18/23)
Carom shot	0.00 (0/0)	0.00 (0/1)
Bank shot	1.00 (3/3)	0.60 (3/5)
Normal pot	0.87 (317/365)	0.90 (317/352)
Fault	0.86 (30/35)	0.88 (30/34)
Others	0.91 (85/93)	0.79 (85/108)
Total	0.88 (453/516)	0.87 (453/523)

• Bank shot: the cue-ball hits the object-ball, and the *object-ball path* intersects one or more cushions.

If the cue stick hits on different parts of the cue-ball, the cue-ball will have different kinds of spins, and will result in various cue-ball paths after hitting the object-ball. The spins of the cue-ball can be classified into two types: forward-backward and clockwise-counterclockwise. The forward-backward type includes a follow shot (the cue-ball spins forward), a draw shot (the cue-ball spins backward), and a stop shot (the cue-ball stops spinning). Fig.3(b) shows the cue-ball paths in the prescribed three kinds of situations. We have six combinations of forward-backward spins and clockwise-counterclockwise spins, like left side follow shot, right side draw shot, etc. Since we know the ball positions in every frame, we can detect all the six hitting skills by the cue-ball trajectory.

We also propose a mechanism to rank the difficulty of a play. The *hitting thickness*, event type, and hitting skill are taken into consideration in the ranking stage.

5. EXPERIMENTAL RESULTS

We evaluate the proposed system based on two series of broadcasting videos, which are 2007 women's pro-billiards competition and 2008 men's pro-billiards competition. The videos are transcoded to MPEG1 format with 640x480 in frame size and 29.97 fps in frame rate. Table 1 shows the performance of the billiards event detection. The event detection module still doesn't comprehensively detect all possible cases in billiards nine-ball videos. How to elaborately distinguish "others" events into finer categorizes will be studied in the future. One possible solution is to take caption information into consideration.

Fig.4 shows a snapshot of the developed billiards wizard system. Users can arbitrarily select a set, go into a game, and see what kinds of events occurred in this game at a glance. The system indicates the ball trajectories of the current play both in the billiards video and on the real-world map (as shown on the right bottom of the frame). The system also shows the corresponding event type, hitting skill, and the rank. Moreover, users can choose every pocket to see the aiming suggestion and position play suggestion.

Fig. 4. A snapshot of the proposed billiards wizard system.

6. CONCLUSIONS

A comprehensive study is conducted to build a billiards tutoring system, including the event detection and the nine-ball play suggestion, to thoroughly tutor users the billiards skills by watching broadcasting nine-ball billiards videos. Our work automatically extracts all play shots by table view detection. We detect and track the balls in each play shot, and apply the technique of camera calibration to obtain the real-world mapping of each ball and the table. We then suggest aiming direction and position play to users, and distinguish each play into six types of events and six types of hitting skills, according to the ball trajectories. Finally, the billiards wizard system successfully combines the results of all our efforts.

7. REFERENCES

[1] A. Hanjalic, "Shot-boundary detection: unraveled and resolved?," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 90–105, 2002.

[2] Y. Liu, S. Jiang, Q. Ye, W. Gao, and Q. Huang, "Playfield detection using adaptive gmm and its applications," *In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 421–424, 2005.

[3] D. Farin, S. Krabbe, P. H. N. de With, and W. Effelsberg, "Robust camera calibration for sport videos using court models," *In Proceedings of SPIE Storage and Retrieval Methods and Applications for Multimedia*, vol. 5307, pp. 80–91, 2004.

[4] M. J. Black and P. Anandan, "A framework for the robust estimation of optical flow," *In Proceedings of Intl Conf. Computer Vision*, pp. 231–236, 1993.

[5] P. Shirley, "Fundamentals of computer graphics," *AK Peters Ltd*, 2002.