J. Vis. Commun. Image R. 23 (2012) 767–781

Contents lists available at SciVerse ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

HMM-based ball hitting event exploration system for broadcast baseball video

Hua-Tsung Chen [a,*], Chien-Li Chou [b], Wei-Chin Tsai [b], Suh-Yin Lee [b], Bao-Shuh P. Lin [a,b]

[a] Information and Communications Technology Lab, National Chiao Tung University, Hsinchu 300, Taiwan
[b] Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan

ARTICLE INFO

Article history:
Received 25 May 2011
Accepted 16 March 2012
Available online 28 March 2012

Keywords:
Multimedia system
Highlight detection
Baseball even recognition
Features extraction
HMM
Scene classification
Sports video analysis
Pattern recognition

ABSTRACT

With the dramatic growth of fandom population, a considerable amount of research efforts have been devoted to baseball video processing. However, little work focuses on the detailed follow-ups of ball hitting events. This paper proposes a HMM-based ball hitting event exploration system for broadcast baseball video. Utilizing the strictly-defined layout of the baseball field, the proposed system first detects the game-specific spatial patterns in the field, such as the field lines, the bases, the pitch mound, etc. Then, the play region—the currently camera-focused region of the baseball field is identified for frame type classification. Since the temporal patterns of presenting the game progress follow a prototypical order, we consider the classified frame types as observation symbols and recognize ball hitting events using HMM. Experiments conducted on broadcast baseball video show encouraging results in frame type classification and ball hitting event recognition. Three practical applications, including highlight clip extraction by user-designated query, storyboard construction, and similar event retrieval, are introduced to address the applicability of our system.

1. Introduction

The explosively increasing amount of digital videos motivates researchers to strive for various aspects of video analysis. In recent years, the amount of multimedia information has grown rapidly. This trend leads to the development of efficient sports video analysis in soccer [1–3], tennis [4–6], basketball [7–9], volleyball [10], baseball [11–24], etc. Automatic sports video analysis has attracted considerable attention, because sport video appeals to large audiences. The possible applications of sports video analysis have been found almost in all sports, among which baseball is a quite popular one. It is time-consuming to watch the whole game video in sequential way, while highlights abstract the game for quick browsing. In addition, highlights can be contributive to tactic inference for coaches, players, and even professional sports fans. For these motivations, we aim at developing a highlight semantics exploring system for the baseball games.

Baseball video is characterized by a strictly-defined structure containing a series of plays and each play starts with a pitch. Hence, PC (pitcher–catcher) shot detection and semantic shot classification play an important role in baseball highlight detection [11,12]. Furthermore, various kinds of pitch analyses have been addressed to derive the correlation between the ball trajectory and the rotation by tracking the translation and rotation of a pitched ball [13], to extract the ball trajectory based on physical characteristics [14], to reconstruct the 3D trajectory of the pitched ball with multiple cameras [15], and even to recognize the pitching style based on the pitcher's posture [16].

Due to broadcast requirement, there has been an essential demand for highlight extraction which aims at abstracting a long game into a compact summary to provide the audience a quick browsing of the game. Moreover, highlight extraction/classification also contributes to many applications such as efficient event indexing and retrieval, providing the reference for tactics inference to the coach and players, user-designated highlight clip extraction, etc. In the past few years, remarkable research has been devoted to baseball video content analysis. Hung and Hsieh [17] categorize shots into pitcher-catcher, infield, outfield, and non-field shots. Combining the detected scoreboard information with the obtained shot types as mid-level cues, Hung et al. use Bayesian Belief Network (BBN) structure for highlight classification. Chu and Wu [18] consider most of the possible conditions in a baseball game based on the game-specific rules and extract the scoreboard information for event detection. Though both Hung and Hsieh [17] and Chu and Wu [18] achieve high accuracy in highlight classification due to the additional information from the scoreboard, their rough shot classification approaches are inadequate to analyze the ball movement and play region transitions for ball hitting events. Gong et al. [19] classify baseball highlights by integrating image, audio, and closed caption cues based on MEM (Maximum Entropy Model). Fleischman et al. [20] use complex temporal features, such as field type, speech, camera motion start time and end time. Temporal data mining techniques are exploited to discover a codebook of frequent temporal patterns for baseball highlight classification.

* Corresponding author. Fax: +886 3 5721490.
E-mail address: huatsung@cs.nctu.edu.tw (H.-T. Chen).

1047-3203/$ - see front matter
http://dx.doi.org/10.1016/j.jvcir.2012.03.006

Because the positions of cameras are fixed in a game and the ways of showing game progressing are similar in different TV channels, each category of semantic baseball event usually has similar scene transitions. For example, a typical "fly out" event can be composed of a PC scene followed by an outfield scene and then an in-grass scene. Hence, the statistical model of HMM is broadly used for highlight detection and classification. Lien et al. [21] extract significant color, object number, motion vector, and player location as features to classify eight semantic scenes: close-up, base, running, pitching, player, infield, outfield, and other. Based on the classified scenes serving as the observation symbol sequence, a 4-state ergodic HMM is applied to detect four baseball events: base hit, ground out, air out, and strike out. Though, good performance is achieved in Lien et al. [21], only four events are detected. It is not so realistic to provide only four events to general users, not to mention the professional players or the coach. Cheng and Hsu [22] fuse visual motion information with audio features, including zero crossing rate, pitch period and Mel-frequency cepstral coefficients (MFCC), to extract baseball highlight based on hidden Markov model (HMM). Mochizuki et al. [23] provide a baseball indexing method based on patternizing baseball scenes using a set of rectangles with image features and a motion vector. Chang et al. [24] assume that most highlights in baseball games consist of certain shot types and these shots have similar transitions in time. Each highlight is described by a HMM and each hidden state is represented by its predefined shot type. Some features are used as observations to train the HMM model for highlight recognition. In Mochizuki et al. [23] and Chang et al. [24], low accuracy and few highlight types are the main disadvantages because the information is too little to detect various highlights and to get high accuracy.

Even if the previous works claim good results on highlight classification, they do not analyze a variety of ball hitting event types and have no idea of the detailed batting process and ball movement within a shot, such as: "The ball batted into the left infield is picked up by an infielder and then thrown to the first baseman." In nature, the first/second/third basemen, the shortstop as well as other players are important objects in terms of event understanding. However, when the camera focuses on a player, it is hard to recognize his fielding position. Hence, in this paper we explore *field shots* (the shots follow the batted ball in the field) and utilize the game-specific spatial patterns, e.g., the bases and the pitch mound, to identify the regions which the ball has passed through. With great success in speech recognition, HMMs are effective models for time-varying patterns and have been used widely in scene modeling for sports video [21–24]. Thus, we propose an HMM-based mechanism to detect and classify up to 11 ball hitting events: (1) single, (2) double, (3) pop up, (4) fly out, (5) ground out, (6) two base hit, (7) foul ball, (8) foul out, (9) double play, (10) home run, and (11) home base out. In addition to providing the detailed description of each play, a baseball exploration system is also developed, so users can efficiently retrieve the batting clips desired. With the proposed framework, highlight extraction and event indexing in baseball video will be more powerful and practical, since comprehensive, detailed, and explicit information about the game can be presented to users.

The rest of the paper is organized as follows. Section 2 describes the system overview of the proposed ball hitting event recognition. The processes of visual feature extraction and frame type classification are explained in Sections 3 and 4, respectively. Section 5 elaborates how to recognize ball hitting events using HMM. Experimental results and discussion are presented in Section 6. Section 7 introduces extensive applications based on the proposed system. Finally, Section 8 concludes this paper and describes the future work.

2. Overview of the proposed HMM-based ball hitting event exploration system

With the foregoing motivation and limitations of the existing works, we develop a HMM-based ball hitting event exploration system for broadcast baseball video. As illustrated in Fig. 1, the system contains three main components including visual feature extraction, frame type classification, and HMM-based ball hitting event recognition. In a baseball game, each play starts with a PC (pitcher–catcher) shot and ends up with specific shots. To trim out the uninteresting segments, e.g., commercials, the pre-processing procedures of shot boundary detection, shot classification, and PC shot detection [11,12,21,25] are required. The feature extraction module extracts significant colors—white, green (grass), and brown (soil), and then recognize the spatial patterns in the baseball field. Fig. 2a shows the full view of a prototypical baseball field, and Fig. 2b shows the spatial patterns to be recognized. Based on the extracted visual features, the system performs frame classification rather than shot classification as in the previous works. The information of the ball movement and play region transitions within a single shot greatly assist the system in comprehending the ball hitting events. Taking the obtained frame types as observation symbols, a HMM-based approach is designed for ball hitting event recognition. Finally, extended applications such as highlight clip extraction by user-designated query, storyboard construction, and similar event retrieval can be implemented based on the proposed scheme.

Compared with the existing works on baseball video analysis, the main contributions of our proposed HMM-based ball hitting event exploration system are summarized as follows. Most of the existing works perform shot classification, and some works are capable of discriminating between infields and outfields at most. However, more explicit information within a field shot should be extracted to comprehend the detailed process of a ball hitting event. With the baseball domain knowledge, we recognize the game-specific spatial patterns and the field layout so as to explore the transition of the play region—the currently camera-focused region of the baseball field. The explicit information of the play region transition significantly facilitates extensive applications.

3. Visual feature extraction

In our proposed system, significant colors and game-specific spatial patterns are extracted as visual features.

3.1. Significant colors

As depicted in Fig. 2, the baseball field is characterized by a well-defined layout of specific colors. Moreover, important lines and the bases are in white color to provide visual assistance for players, umpires, and audience. Therefore, color is an effective visual feature in baseball video analysis, especially the significant colors: white, green (grass), and brown (soil). The in-frame color of each baseball game might vary due to different viewing angles and lighting conditions. We propose to define dominant colors by histogram. To obtain the color distribution of *grass* and *soil* in video frames, several baseball clips from different video sources are input to compute the color histograms in RGB and HSI (Hue-Saturation-Intensity) color spaces. Fig. 3 shows the color histograms of baseball clips from different sources. The hue value in HSI color space is adequate to define the dominant colors for two reasons: (1) we have observed that the hue value is relatively stable within a single game despite the lighting conditions, and (2) the hue value has good discrimination since the grass and soil colors form salient peaks in the hue histogram.

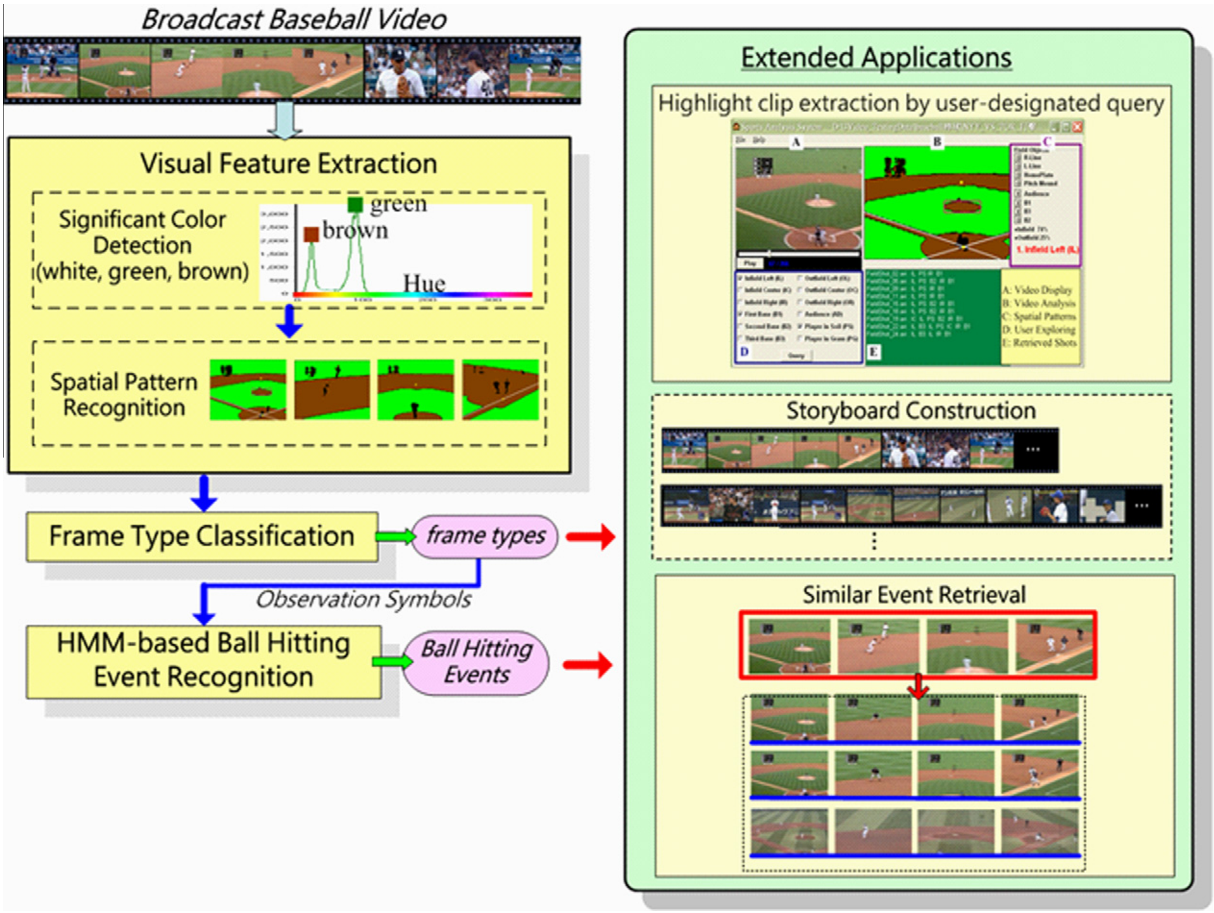

Fig. 1. Flowchart of the proposed HMM-based ball hitting event exploration system.

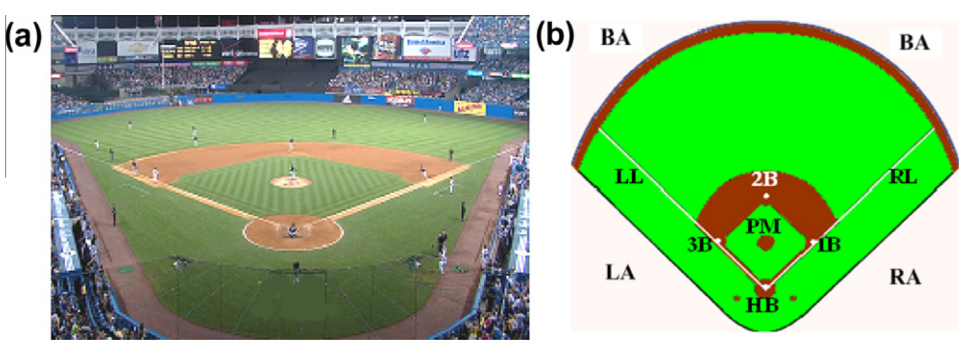

Fig. 2. Prototypical baseball field: (a) full view of a real baseball field; (b) illustration of spatial patterns.

In a field shot, the initial frames mainly contain the baseball field, while the later frames, which might zoom in on a player or move to the audience, contain less proportion of the field. Therefore, it is reasonable to compute the color histogram from the initial frames of a field shot and define the grass and soil colors. Fig. 4 demonstrates the spatial distribution of significant colors. Fig. 4a shows a field frame. In the hue histogram of Fig. 4b, significant colors can be defined: the peak of small hue value representing the soil color and the peak of large hue value representing the grass color. The regions segmented by significant colors are shown in Fig. 4c, where grass regions are shown in green, soil regions in brown, and others in black. The pixels of high intensity values are detected as white pixels, as presented in Fig. 4d.

3.2. Spatial patterns

With the extracted significant colors (grass, soil, and white), we are ready to analyze the field shots and detect the spatial patterns: left line *LL*, right line *RL*, pitch mound *PM*, home base *HB*, first base *1B*, second base *2B*, third base *3B*, back auditorium *BA*, left auditorium *LA*, and right auditorium *RA*. Please refer to Fig. 2b. Figs. 5 and 6 exemplify the detection of spatial patterns. The detailed detection processes are elaborated as follows. For clarity, the names of the spatial patterns are abbreviated in *italic* type.

3.2.1. Field lines: left line LL and right line RL

For visual clarity, the field lines and important markers are in white color, as specified in the official game rules. However, there

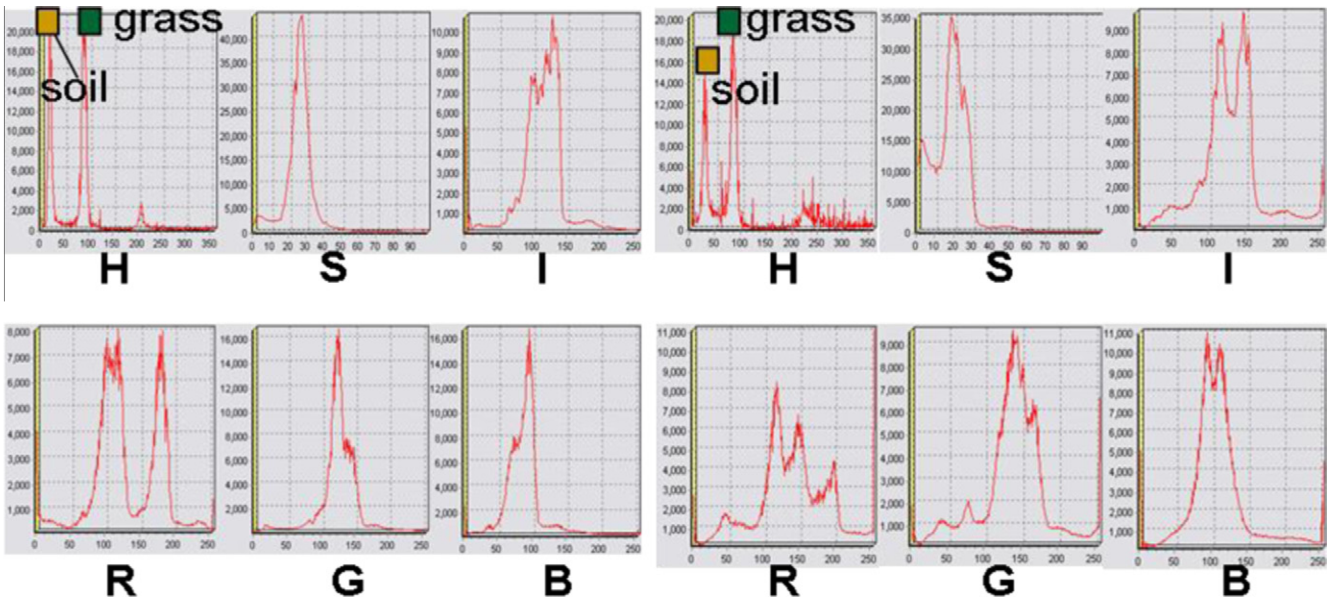

Fig. 3. Color space of RGB and HSI of baseball clips from different sources. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

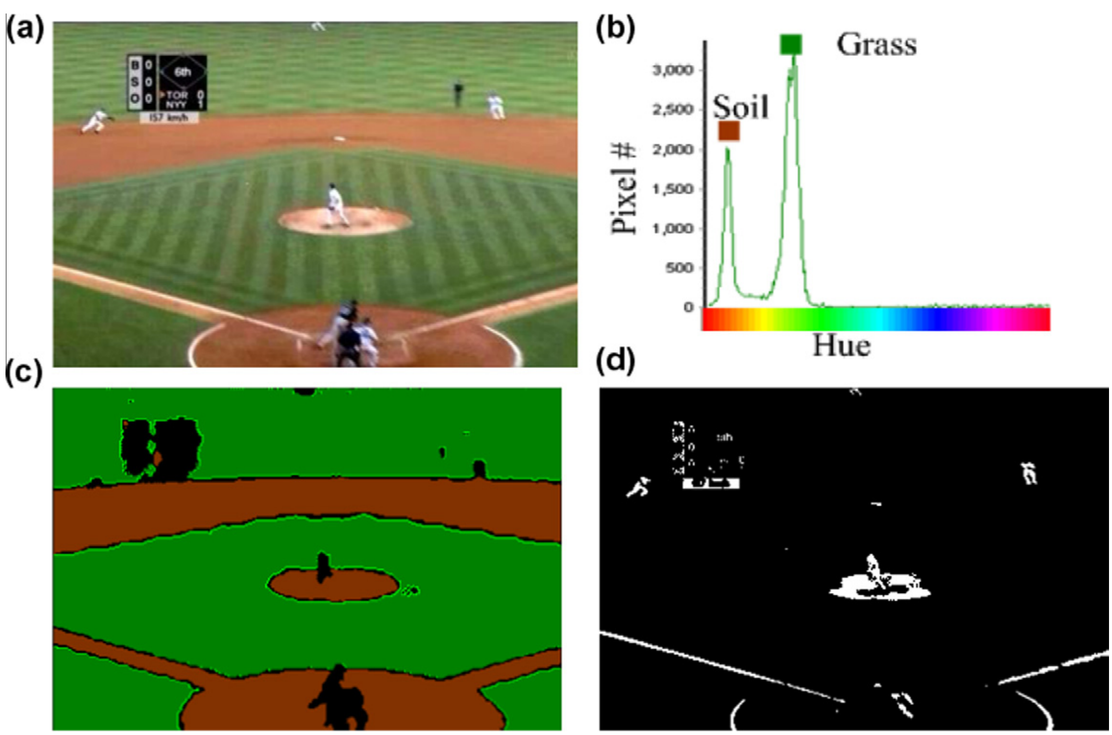

Fig. 4. Spatial distribution of significant colors: (a) field frame; (b) hue histogram; (c) segmented regions: grass regions shown in green, soil regions in brown, and others in black; (d) detected white pixels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

may be other white objects in frames such as advertisement logos, the uniforms of the players, and the clothes of the audience. Hence, additional criteria and constrains are applied to white line pixel detection [26,27]. As illustrated in Fig. 7, each square represents one pixel and the central one drawn in gray is a candidate pixel. Assuming that white lines are typically no wider than τ_d pixels ($\tau_d = 4$ in our system), we check the brightness of the four pixels, marked '●' and '○', at a distance of τ_w pixels away from the candidate pixel on the four directions. The central candidate pixel is identified as a white line pixel only if both pixels marked '●' or both pixels marked '○' are with lower brightness than the candidate pixel. This process prevents most of the pixels in white regions or white uniforms being detected as white line pixels.

Furthermore, we apply the line-structure constraint [26] to exclude the white pixels in finely textured regions. The structure matrix S [28] computed over a small window of size $2b+1$ (we use $b = 2$) around each candidate pixel (p_x, p_y), as defined in Eq. (1), is used to recognize texture regions, where $I(x,y)$ represents the intensity component in HSI color space and $\nabla I(x,y)$ is the image gradient.

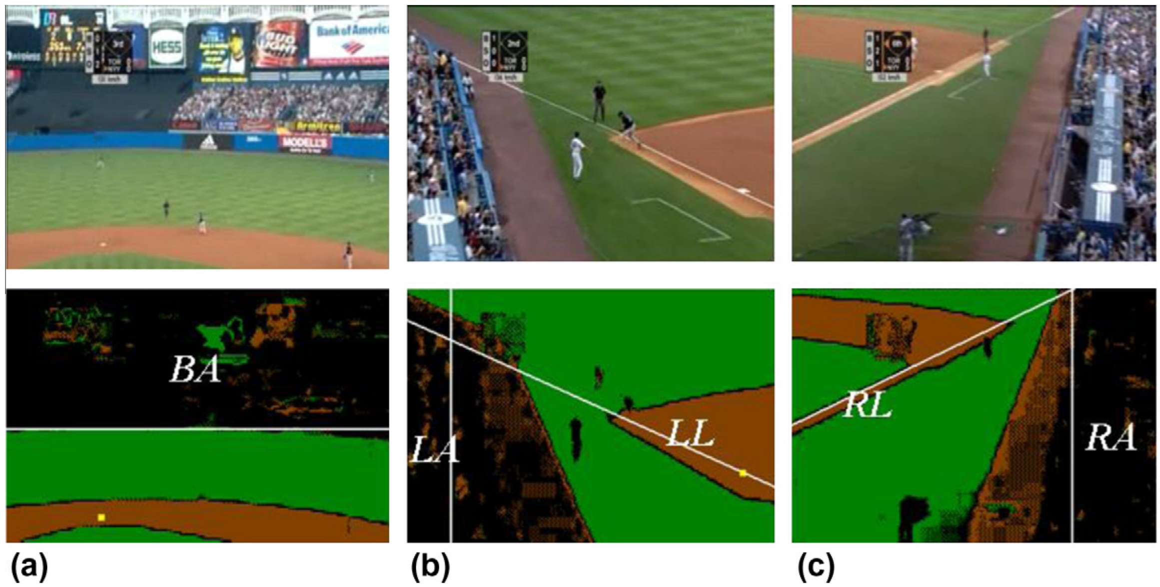

Fig. 5. Detection of spatial patterns: (a) back auditorium BA; (b) left line LL and left auditorium LA; (c) right line RL and right auditorium RA.

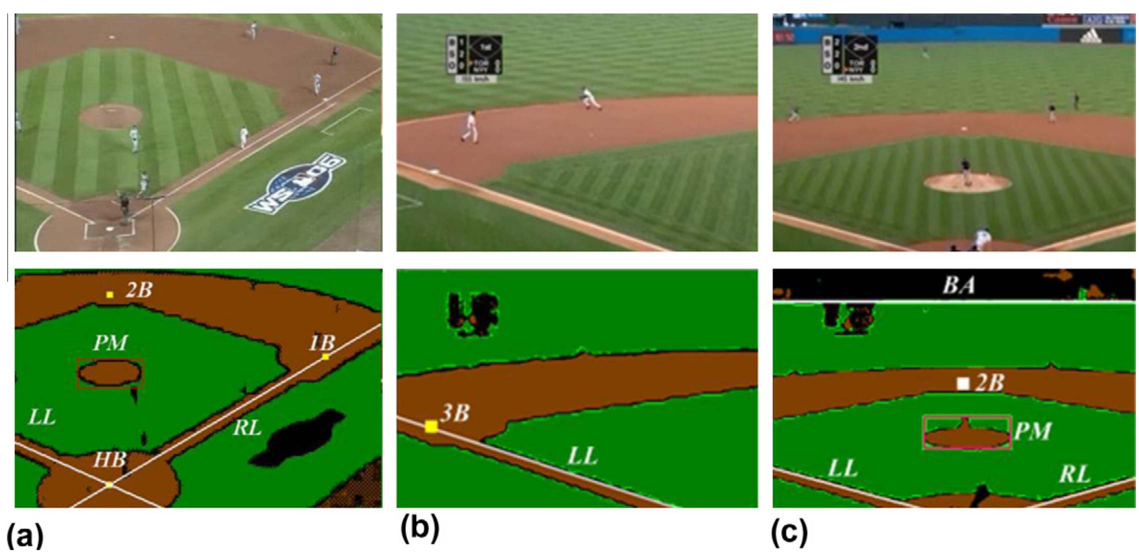

Fig. 6. Examples of spatial pattern detection.

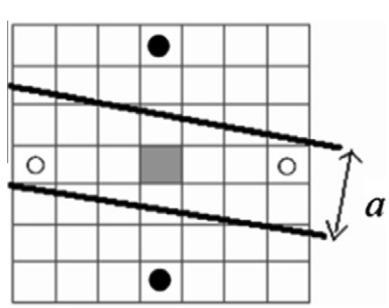

Fig. 7. Illustration of part of an image containing a white line.

$$S = \sum_{x=P_x-b}^{P_x+b} \sum_{y=P_y-b}^{P_y+b} \nabla I(x,y) \cdot (\nabla I(x,y))^T \tag{1}$$

Depending on the two eigenvalues of matrix S, called λ_1 and λ_2 ($\lambda_1 \geqslant \lambda_2$), the texture can be classified into *textured* (λ_1, λ_2 are large), *linear* ($\lambda_1 \gg \lambda_2$), and *flat* (λ_1, λ_2 are small). On the straight field lines, the *linear* case will apply to retain the white pixels only if $\lambda_1 > a\lambda_2$ ($a = 4$ in our system). Fig. 8 demonstrates the sample result of white line pixel detection. The original frames are presented in Fig. 8a. Fig. 8b shows the high intensity pixels before white line pixel detection. With the line-structure constraint, Fig. 8c shows that white line pixel candidates are retained only if the pixel neighbor shows a linear structure and the number of false detections is reduced.

With the white line pixels detected, a growing algorithm, which produces a vector representation of the line segments [29], is applied to the extracted white pixels. Field lines *LL* and *RL* are then obtained by joining together the line segments which are close and collinear, as shown in Figs. 5b, c and 6.

3.2.2. Left auditorium LA, right auditorium RA, and back auditorium BA

The left, right, and top areas which contain high texture and no dominant colors are considered as the spatial patterns *LA*, *RA*, and

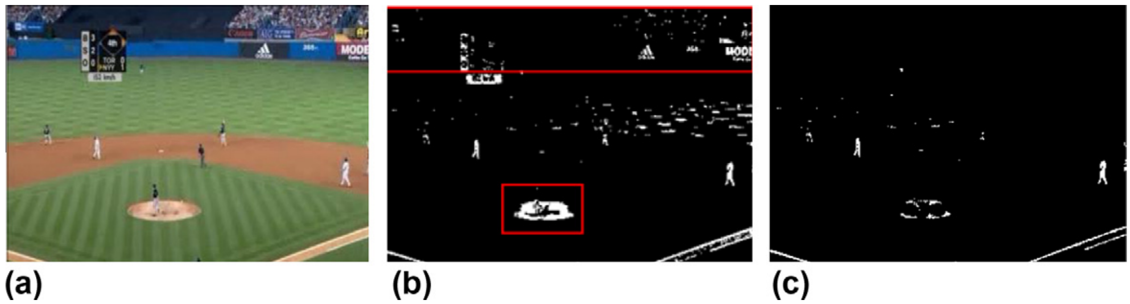

Fig. 8. Sample result of white line pixel detection: (a) original frame; (b) high intensity pixels; (c) white line pixel candidates with the line-structure constraint.

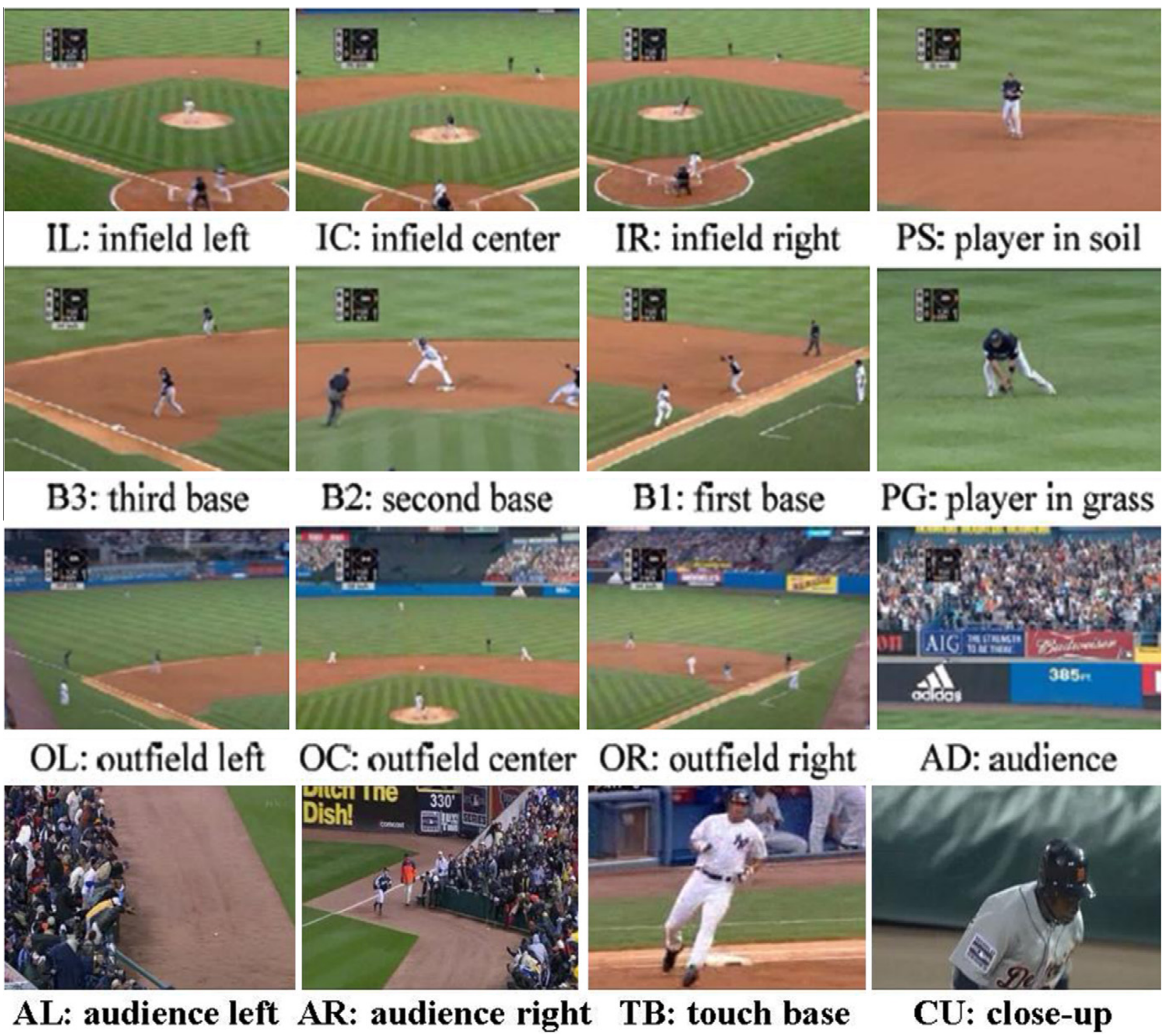

Fig. 9. Sixteen typical frame types.

BA, respectively, as exemplified in Fig. 6. In Fig. 5a, the black area above the white horizontal line is detected as *BA*. In Fig. 5b and c, left and right black areas outside the vertical lines are the detected *LA* and *RA*, respectively.

3.2.3. Pitch mound PM

An ellipse soil region surrounded by a grass region is recognized as *PM* as shown in Fig. 6a and c. Some constraints are used reduce false detections: (1) *PM* cannot be on *LL* or *RL*; (2) PM should be on the left/right side of *RL*/*LL*, if detected; (3) there should not be large difference in the number of soil pixels between the top half region and the bottom half region of the detected soil ellipse; (4) there should not be large difference in the number of soil pixels between the left half region and the right half region of the detected soil ellipse.

3.2.4. Home base HB

HB is located at the intersection of *LL* and *RL* if both field lines are detected, as shown in Fig. 6a.

3.2.5. First base 1B and third base 3B

The white square region located on *RL*, if detected, in soil region is identified as *1B*, as shown in Fig. 6a. Similarly, the white square region located on *LL*, if detected, in soil region is identified as *3B*, as shown in Fig. 6b.

3.2.6. Second base 2B

In a soil region, a white square region on neither field line is identified as *2B*, as shown in Fig. 6a and c.

4. Frame type classification and annotation string generation

In order to comprehend the detailed process of the ball hitting event, we have to recognize the *play region*, the currently camera-focused region in the baseball field, for frame type classification. Based on the detected spatial patterns, we classify each field frame into one of the 16 types: IL (infield left), IC (infield center), IR (infield right), B1 (first base), B2 (second base), B3 (third base), OL (outfield left), OC (outfield center), OR (outfield right), PS (player in soil), PG (player in grass), AD (audience), AL (audience left), AR (audience right), TB (touch base), and CU (close-up), as shown in Fig. 9. Note that B1, B2, and B3 here represent "frame types" while 1*B*, 2*B*, and 3*B* in Section 3.2 represent "spatial patterns."

With the baseball domain knowledge, we define explicit rules for frame type classification, as listed in Table 1. The function P(*A*) returns the percentage of the area *A* in a frame, and E(*S*) returns whether the spatial pattern *S* exists or not. In addition, we tri-partition a frame into *left*, *center*, and *right* parts. L(*S*) returns which part the spatial pattern *S* is located in. The thresholds $\tau_1 \sim \tau_5$ are determined by training. We ask an experienced baseball expert to watch 122 training clips and label the frame types through a simple user interface. The thresholds $\tau_1 \sim \tau_5$ are determined by seeking for the values which best classify the frame types in the training data.

Each field frame is classified into one of the 16 types by applying the rules on the spatial patterns. Take IL (infield left) as an example. A field frame would be identified as IR under the following conditions:

(1) The percentage of *BA* area in a frame is no more than $\tau_{1\%}$, *PM* exists and is located at the *left* one-third of a frame).
(2) The percentage of *BA* area in a frame is no more than $\tau_{1\%}$, *PM* does not exist, *RL* exists, and *1B* does not exist.
(3) The percentage of *BA* area in a frame is no more than $\tau_{1\%}$, *PM* does not exist, *RL* exists, *1B* exists, and the percentage of soil area is no more than $\tau_{2\%}$.

The scheme of frame type classification within a field shot is illustrated in Fig. 10. The spatial patterns are first detected by the distribution of significant color pixels in field frames. According to the rules on the spatial patterns, each field frame is then classified into one of the 16 typical play region types. To filter out instantaneous misclassifications of frame types, a fixed length temporal window and majority voting are applied. Thus, an *annotation string* which describes the transition of play regions contained in a field shot can be generated. The content of the sample field shot in Fig. 10 says that the ball is first batted into the left infield. Then, the shortstop picks up the ball and throws it to the first baseman. The batting process can be appropriately abstracted by the output annotation string: IL (infield left) → PS (player in soil) → IR (infield right) → B1 (first base).

Table 1
Rules of frame type classification.

IR:	$\{P(BA) \leqslant \tau_{1\%}, E(PM), L(PM) = left\} \|\|$ $\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(RL), \sim E(1B)\} \|\|$ $\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(RL), E(1B), P(soil) \leqslant \tau_{2\%}\}$
IC:	$\{P(BA) \leqslant \tau_{1\%}, E(PM), L(PM) = center\} \|\|$ $\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), \sim E(RL), \sim E(LL), E(2B), P(soil) \leqslant \tau_{2\%}\}$
IL:	$\{P(BA) \leqslant \tau_{1\%}, E(PM), L(PM) = right\} \|\|$ $\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(LL), \sim E(3B)\} \|\|$ $\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(LL), E(3B), P(soil) \leqslant \tau_{2\%}\}$
B1:	$\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(RL), E(1B), P(soil) > \tau_{2\%}\}$
B2:	$\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), \sim E(RL), \sim E(LL), E(2B), P(soil) > \tau_{2\%}\}$
B3:	$\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), E(LL), E(3B), P(soil) > \tau_{2\%}\}$
OR:	$\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, E(PM), L(PM) = left\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), E(2B), L(2B) = left\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), \sim E(2B), E(RL), \sim E(LL)\}$
OC:	$\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, E(PM), L(PM) = center\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), E(2B), L(2B) = center\}$
OL:	$\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, E(PM), L(PM) = right\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), E(2B), L(2B) = right\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), \sim E(2B), E(LL), \sim E(RL)\}$
PS:	$\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), \sim E(2B), \sim E(RL), \sim E(LL), P(soil) > \tau_{2\%}\}$
PG:	$\{\tau_{1\%} < P(BA) \leqslant \tau_{3\%}, \sim E(PM), \sim E(2B), \sim E(RL), \sim E(LL)\} \|\|$ $\{P(grass) > \tau_{4\%}, \sim E(PM), \sim E(2B), \sim E(RL), \sim E(LL)\}$
AD:	$\{P(BA) > \tau_{3\%}\}$
AR:	$\{\tau_{1\%} < P(RA) \leqslant \tau_{5\%}\}$
AL:	$\{\tau_{1\%} < P(LA) \leqslant \tau_{5\%}\}$
TB:	$\{\tau_{1\%} < P(BA) \leqslant \tau_{5\%}, P(soil) > \tau_{2\%}, E(RL), E(1B)\} \|\|$ $\{\tau_{1\%} < P(BA) \leqslant \tau_{5\%}, P(soil) > \tau_{2\%}, E(LL), E(3B)\}$
CU:	$\{P(BA) \leqslant \tau_{1\%}, \sim E(PM), \sim E(2B), \sim E(RL), \sim E(LL), P(grass) + P(soil) < \tau_{5\%}\}$
Unknown: others	

5. HMM-based ball hitting event recognition

This main objective of this paper is to develop a ball hitting event exploration system to trace the play region transition and recognize the ball hitting event. Regarding the classified frame types as the observation symbols, we propose a HMM-based approach to recognize various ball hitting events, including: single, double, pop up, fly out, ground out, two-base out, foul ball, foul out, double play, home run, and home base out.

Generally, HMM is expressed by a 3-tuple parameters $\lambda = \{A, B, \pi\}$. The segmental K-means algorithm is used to create initial HMM parameters and the standard Baum–Welch algorithm is used to optimize the model parameters [30]. The conventional implementation issues in HMM include: (1) number of states, (2) initialization, and (3) distribution of observation at each state. The essential HMM elements of our proposed system are summarized as follows.

State S: The state numbers are selected empirically depending on different baseball events.

Observation O: The classified frame types are taken as the observation symbols.

Observation distribution matrix B: We use K-means algorithm and choose the Gaussian distribution at each state [31].

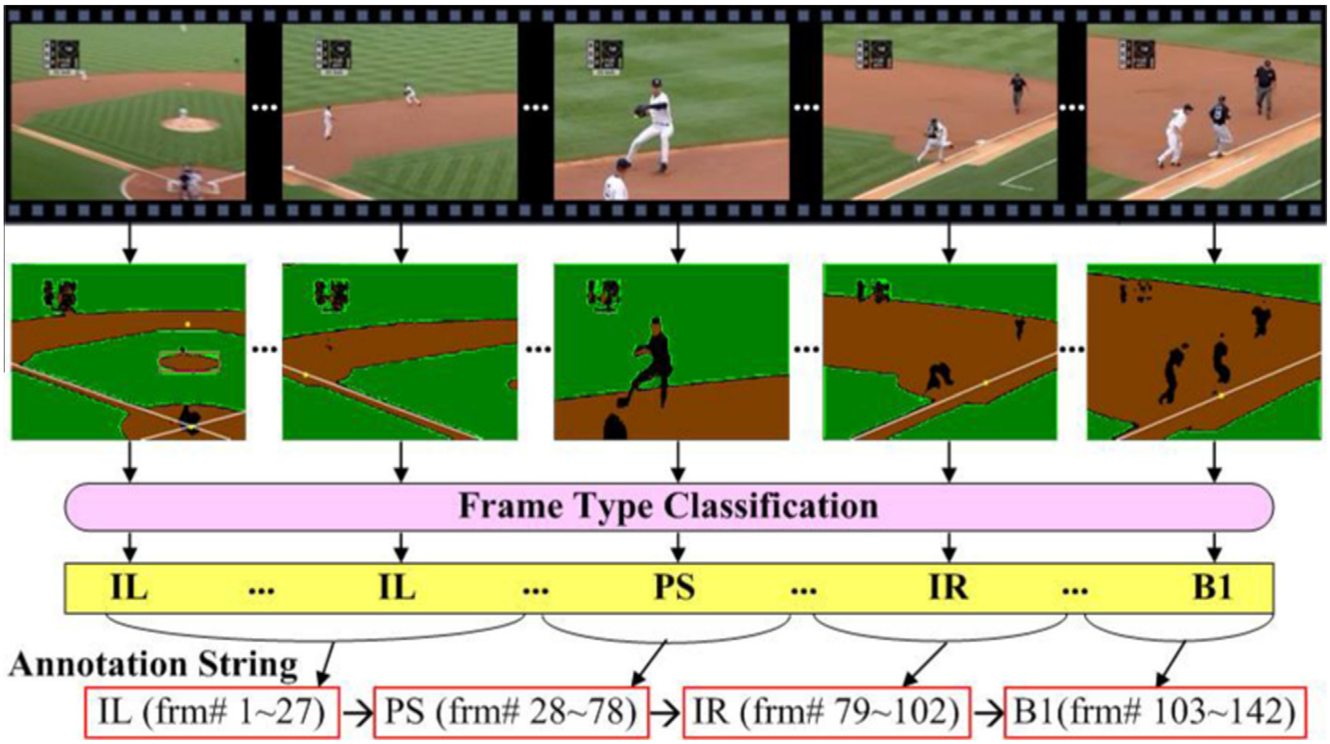

Fig. 10. Illustration of frame type classification and annotation string generation.

Transition probability matrix A: The state transition probability can be learned by the segmental K-means algorithm.

Initial state probability matrix π*:* The probability of occurrence of the first state is initialized by segmental K-means algorithm after determining the number of states.

The idea behind using the HMMs is to construct a model for each ball hitting event that we want to recognize. HMMs give a state based representation for each event. Based on the classified frame types serving as the observation symbol sequence O, the parameters $\{A, B, \pi\}$ for HMM are estimated using the Baum–Welch algorithm [30]. Given the observation symbol sequence $O = [o_1, o_2, \ldots, o_W]$, the observation probability $P(O|\lambda)$ for each ball hitting event is computed via the forward–backward procedure [30]. A forward variable $\alpha_t(i)$ is defined to compute the probability of partial observing sequence of state i at time t for the model λ denoted as $P(o_1 o_2 \ldots o_W, q_W = i|\lambda)$. $P(O|\lambda)$ is computed as follows.

$$\text{(a) Initialization: } \alpha_t(i) = \pi_i b_i(o_1) \tag{2}$$

$$\text{(b) Induction: } \alpha_{t+1} = \left[\sum_{i=1}^{N} \alpha_w(i) a_{ij}\right] b_j(o_w + 1), 1 \leqslant w \leqslant W - 1 \tag{3}$$

$$\text{(c) Termination: } P(O|\lambda) = \sum_{i=1}^{N} \alpha_W(i) \tag{4}$$

Finally, we can then recognize the ball hitting event via finding the model with the highest probability.

6. Experimental results and discussion

To demonstrate the effectiveness of the proposed frame type classification and ball hitting event recognition approaches, we conduct the experiments on the video data of MLB (Major League Baseball) and JPB (Japanese Professional Baseball) games. In total, we have 253 video clips recorded from live broadcast television programs and compressed in MPEG-2 video standard with frame resolution of 352×240 (29.97 fps). For the evaluation of our proposed methods, 122 clips are randomly selected for training and the other 131 clips are for testing.

6.1. Frame type classification

In order to comprehend the ball hitting event content and the region transition, we recognize the play region of each frame based on the detected spatial patterns. Each frame is classified into one of the 16 frame types: IL (infield left), IC (infield center), IR (infield right), B1 (first base), B2 (second base), B3 (third base), OL (outfield left), OC (outfield center), OR (outfield right), PS (player in soil), PG (player in grass), AD (audience), AL (audience left), AR (audience right), TB (touch base), and CU (close-up). Please refer to Fig. 9 for the 16 frame types. The experimental result of frame type classification is presented in the Table 2, where the column "total" indicates the total times of the frame type (designated in the first column) appearing. There are eight *unknowns*, which are regarded as *misses* in frame type classification. Note that a shot does not comprise only one frame type. The ground-truth frame types contained in each shot are manually identified. The "correct" and "false alarm" represent the number of correct detections and false alarms. The precision and recall are defined by Eqs. (5) and (6).

$$precision = \frac{\#correct}{\#correct + \#false\ alarm} \tag{5}$$

$$recall = \frac{\#correct}{\#correct + \#miss}, (\#correct + \#miss = \#total) \tag{6}$$

As in Table 2, the obtained precision and recall rates are about 90%, except for the precision rates of B2 (second base) and AD (audience), and the recall rates of B2 (second base), PS (player in soil), and TB (touch base). By inspecting the experimental process, we have some observations. The low precision and recall rates of frame type B2 mainly result from the incorrect detection or false alarm of the spatial pattern *2B*. For example, the ball in the soil region may be detected as *2B*, as shown in Fig. 11. Moreover, the

Table 2
Performance of frame type classification.

Frame type	Total	Correct	False alarm	Precision (%)	Recall (%)
IL	66	60	5	92.3	90.9
IR	112	102	3	97.1	91.1
IC	92	84	4	95.5	91.3
OL	57	52	6	89.7	91.2
OR	81	78	6	92.9	96.3
OC	76	68	9	88.3	89.5
B1	323	306	6	98.1	94.7
B2	62	51	8	86.4	82.3
B3	43	42	5	89.4	97.7
PG	513	489	45	91.6	95.3
PS	381	328	35	90.4	86.1
AD	97	89	15	85.6	91.8
AR	84	84	4	95.5	100
AL	73	73	5	93.6	100
TB	84	68	5	93.2	81.0
CU	129	119	11	91.5	92.2
Overall	2273	2093	172	92.4	92.1

missed detection of the spatial pattern *B2* will cause a 2B frame to be classified into the frame type PS. We should set a search window for *B2*. However, it is difficult to decide the threshold of the window size since the size of *B2* varies in frames due to the camera zooming. The proposed system may confuse TB (touch base) and AD (audience) because a TB frame may contain high texture and no dominant colors, just like an AD frame. These problems can be improved by enhancing spatial pattern detection, refining the rules of frame type classification, and even adding the procedure of player detection. Overall, the proposed system achieves good performance in frame type classification, which facilitates the subsequent analyses.

6.2. Ball hitting event recognition

HMMs give a state-based representation for each ball hitting event which we want to recognize. Based on the classified frame types regarded as the observation symbols, we apply an HMM-based approach to recognize 11 ball hitting events, including: single, double, pop up, fly out, ground out, two-base out, foul ball, foul out, double play, home run, and home base out. The performance of ball hitting event recognition is presented in the Table 3, where the terms "total," "correct," "false alarm," "precision," and "recall" have the same meanings as in Section 6.1.

As shown in Table 3, the propose system is able to accurately recognize most of the ball hitting events, except for "double," "foul out," "double play," and "two-base out." The low recall rates of "double" and "two-base out" are mainly caused by the incorrect classification of frame type B2. In addition, a "double" event and a "home run" event have almost the same transitions of play regions in the case of the batter hitting the ball to the auditorium wall. More ambiguous cases are discussed and illustrated in the

Table 3
Performance of ball hitting event recognition.

Event type	Total	Correct	False alarm	Precision (%)	Recall (%)
1. Single	25	20	4	83.3	80.0
2. Double	8	2	1	66.7	25.0
3. Pop up	7	7	2	77.8	100
4. Fly out	22	18	6	75.0	81.8
5. Foul out	4	2	0	100	50.0
6. Ground out	29	27	4	87.1	93.1
7. Two-base out	4	2	0	100	50.0
8. Foul ball	18	18	3	85.7	100
9. Double play	4	4	2	66.7	100
10. Home run	6	5	1	83.3	83.3
11. Home base out	4	3	0	100	75.0
Overall	131	108	23	82.4	82.4

following. In Fig. 12, a "ground out" event and a "double play" event have almost the same transitions of play regions when the batter hits the ball toward the second base. Fig. 13 shows an example of incorrect recognition between "foul ball" and "home run" due to the quite similar transitions of play regions. Fig. 14a–c shows the three events: "ground out," "foul ball," and "single," respectively. However, the patterns of play region transition and camera motion are essentially the same in these cases. Actually, the players perform almost the same actions in the three cases: the batter runs to the first base, a fielder catches the ball and throw it to the first baseman. The only difference is that the umpire judges the ball hitting result as "ground out," "foul ball," or "single" according to what he has seen, subjectively and empirically.

In summary, the reasons causing errors in ball hitting event recognition include: (1) similar shot transitions, (2) incorrect spatial pattern detection, and (3) ambiguity in umpire judgment. These problems could be overcome by detecting the ball and players or involving additional cues such as the scoreboard information. So far, we obtain encouraging experimental results and achieve an acceptable performance of the average precision and recall rates above 80%.

6.3. Comparison with existing algorithms of baseball event classification

In the literature, many approaches of baseball event classification have been developed. For performance comparison, two existing works including Lien's HMM-based event classification [21] and Fleischman's temporal feature induction (TFI)-based method [20] are implemented and evaluated using the same data set, except for the "foul ball" cases, which are not dealt with in Lien et al. [21] and Fleischman et al. [20]. Besides, our testing data set contains four "foul out" cases. They are regarded as the "air out" class when evaluating [21], and for [22] three of them belong to the "infield out" class and the other one is "outfield out." On the other hand, our testing data set does not include the "strike out" and "walk" events,

Fig. 11. Ambiguity in the spatial pattern 2B (second base).

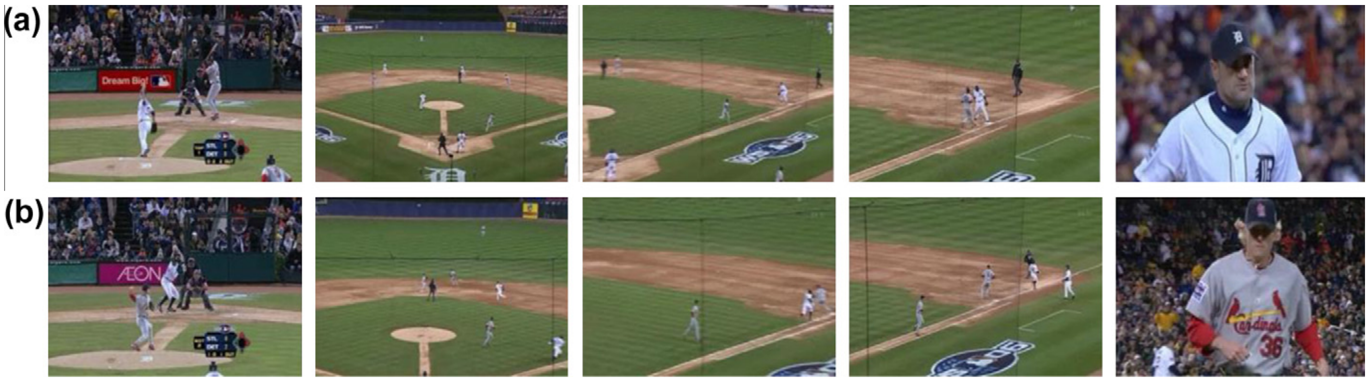

Fig. 12. Similar play region transitions in (a) ground out and (b) double play.

Fig. 13. Comparison between (a) foul ball and (b) home run.

Fig. 14. Ambiguous cases of (a) ground out, (b) foul ball, and (c) single.

because our proposed system mainly focuses on the events when the ball is hit out by the batter (the so-called "ball hitting events").

With Lien's work [21], the boundaries of 442 shots (out of the total 464 shots) are corrected detected, and the accuracy rate (*#correct*/*#total*) of shot boundary detection is 95.3%. Eight scene types including pitching, base, running, other, close-up, player, infield, and outfield are classified using the features of global motion, color distribution, and object information. As presented in Table 4, the experimental results show that all the pitching scenes can be correctly detected (recall rate = 100%) with a precision rate of

Table 4
Scene classification results of Lien et al. [21].

Scene type	Total	Correct	False alarm	Precision (%)	Recall (%)
1. Pitching	113	113	12	90.4	100
2. Base	19	13	5	72.2	68.4
3. Running	44	38	17	69.1	86.4
4. Other	42	33	24	57.9	78.6
5. Close-up	101	71	4	94.6	70.3
6. Player	17	15	3	83.3	88.2
7. Infield	69	64	0	100	92.8
8. Outfield	59	47	5	90.4	79.7
Overall	464	394	70	84.9	84.9

Table 5
Performance of HMM-based event classification of Lien et al. [21].

Event type	Total	Correct	False alarm	Precision (%)	Recall (%)
1. Base hit	41	27	8	77.1	65.9
2. Ground out	37	37	16	69.8	100
3. Air out	35	20	5	80.0	57.1
4. Strike out	0	–	–	–	–
Overall	113	84	29	74.3	74.3

Table 6
Performance of TFI-based event classification of Fleischman et al. [20].

Event type	Total	Correct	False alarm	Precision (%)	Recall (%)
1. Home run	6	3	3	50.0	50.0
2. Outfield hit	30	18	18	50.0	60.0
3. Outfield out	25	12	11	52.2	48.0
4. Infield hit	5	1	0	100	20.0
5. Infield out	47	38	9	80.9	80.9
6. Strike out	0	–	–	–	–
7. Walk	0	–	–	–	–
Overall	113	72	41	63.7	63.7

90.4%, and the overall precision and recall rates of scene classification are about 85%. Despite the satisfying results of scene classification, the average precision and recall rates of baseball event classification for [21] are about 74%, as shown in Table 5. The critical factor causing errors in the event classification is that only one key-frame extracted for each video shot is insufficient. A field shot following the ball batted out may contain more than one scene. The scene transitions within a shot bring significant information, but if only one key-frame is extracted for the shot, much information may be neglected. This is also why our proposed system performs *within-shot* frame type classification.

In Fleischman's temporal feature induction (TFI)-based method [20], temporal patterns are mined from the low-level features of scene types (pitch/field/other), camera motions (pan/tilt /zoom), and sound classes (speech/cheer/music) for baseball event classification. The maximum depth of the temporal pattern mined is set to five, which is verified by Fleischman et al. [20] to result in a peak performance. The results of the TFI-based baseball even classification of Fleischman et al. [20] are presented in Table 6, wherein the average precision and recall rates are about 64%. Compared with our proposed system, Fleischman's work adds audio features and exams the temporal relations among features. However, only three scene types (pitch/field/other) used in Fleischman et al. [20] seem unable to bring sufficient information to classify various baseball events effectively. Overall speaking, our proposed system has the advantage of extracting the information of the ball movement and scene transitions within a single shot, which significantly assists in classifying various ball hitting events. The experimental results and comparison indicate that our proposed method outperforms Lien's HMM-based event classification [21] and Fleischman's temporal feature induction (TFI)-based method [20].

7. Applications

7.1. Highlight clip extraction by user-designated query

We have implemented a preliminary prototype of the user interface of the proposed baseball exploration system, as shown in Fig. 15. The video is displayed in area **A** and the visual presentation of the video analysis is provided in **B**. Area **C** gives the information about the detected spatial patterns. Furthermore, users are allowed to designate play region types in **D** for exploration. The highlight clips containing the user-designated play region types are retrieved and listed in **E** with their respective annotation strings.

7.2. Storyboard production

To quickly browse numerous baseball video clips, a *storyboard* which provides a concise video content representation based on the video content would be really appreciated. Fig. 16 shows some storyboard examples of baseball games. A storyboard allows the users to have an idea of the video content without having to watch the video entirely. Recently, storyboard production has been the goal of the so-called *video summarization* techniques, which compute the difference between frames and/or the importance of each frame based on visual features for extracting the relevant frames. In this section, we present an approach to produce compact and complete storyboards with high expressiveness and information for baseball video clips based on the annotation strings generated in Section 4.

Storyboard production involves an important task: the selection of the relevant frames to be displayed. Since a pitch shot has little camera motion, the frames in a pitch shot are similar to each other. Thus, no matter which frame is chosen as the relevant frame for storyboard production, users are able to perceive that the shot is to convey the pitch action (please see the leading picture of each row in Fig. 16). Similarly, users can have an idea that a shot is to present the overview of the audience or the close-up of the pitcher, batter or couch after seeing one of the frames in the idle shot. Hence, for computational simplicity, we select the first frame of a pitch shot or an idle shot as the relevant frame to be displayed in the storyboard.

However, in a field shot, the camera tends to follow the ball moving in the field. Only one frame cannot provide adequate information for users to comprehend the route of the ball batted out. How to select as few relevant frames in a field shot as possible but to provide adequate information is the major problem we aspire to work out. In Section 4, we classify the frame types based on the visual features and spatial patterns in the baseball field. Thus, we can further divide a field shot into sub-shots in each of which the frames have the same play region type. We can say that the frames in a sub-shot are similar semantically and visually, since they contain the same spatial patterns and have similar visual features. Hence, only one frame for each sub-shot needs to be displayed in the storyboard. Here, we select the middle frame in a sub-shot as the relevant frame because the middle frame is distinct from the frames in the neighboring sub-shots while the frames close to the boundaries of the sub-shot might be similar to the frames in the neighboring sub-shots. Take Fig. 10 for example. The play regions appearing in the field shot are IL (infield left, frames #1–27), PS (player in soil, frame #28–78), IR (infield right, frames #79–102), and B1 (first base region, frames #103–142). Thus, the field shot is divided into four sub-shot, and the frames

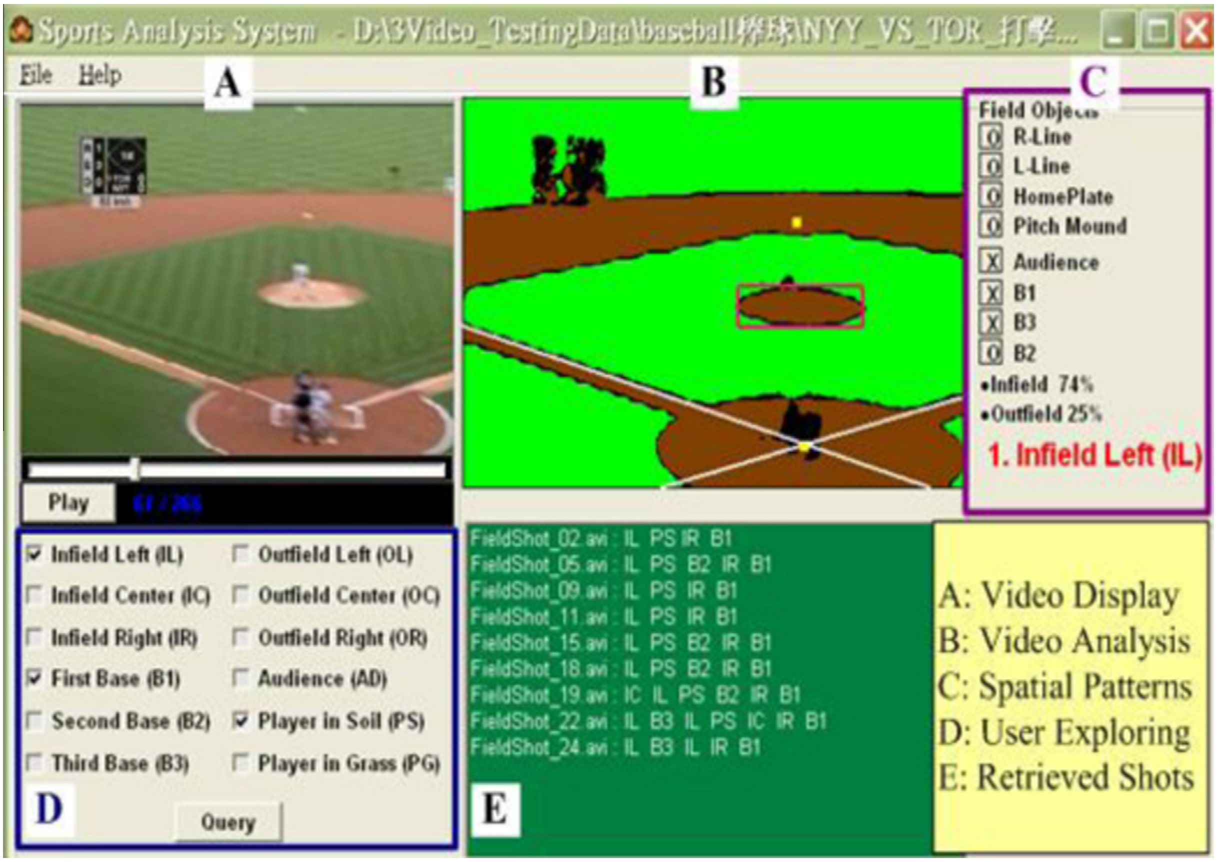

Fig. 15. User interface of the proposed baseball exploration system.

Fig. 16. Storyboards of baseball games.

#14, #53, #90, and #122 are selected to be displayed in the storyboard. In this way, we are able produce a storyboard which uses as few frames as possible to provide adequate information and convey the video content.

7.3. Similar event retrieval

Usually, after viewing a highlight clip, users tend to view some other similar or relevant highlight clips. Furthermore, baseball fans and professionals may have interests in some special events, highlights or specific defense patterns. They would like to retrieve similar ball hitting events from different games for viewing and comparison. In this section, we propose an effective algorithm to retrieve similar ball hitting events based on the proposed spatial pattern detection and frame type classification method.

Similar event retrieval involves two aspects: one is the choice of representation for the data and the other is the definition of similarity measurement. In Section 4, we present a method to classify

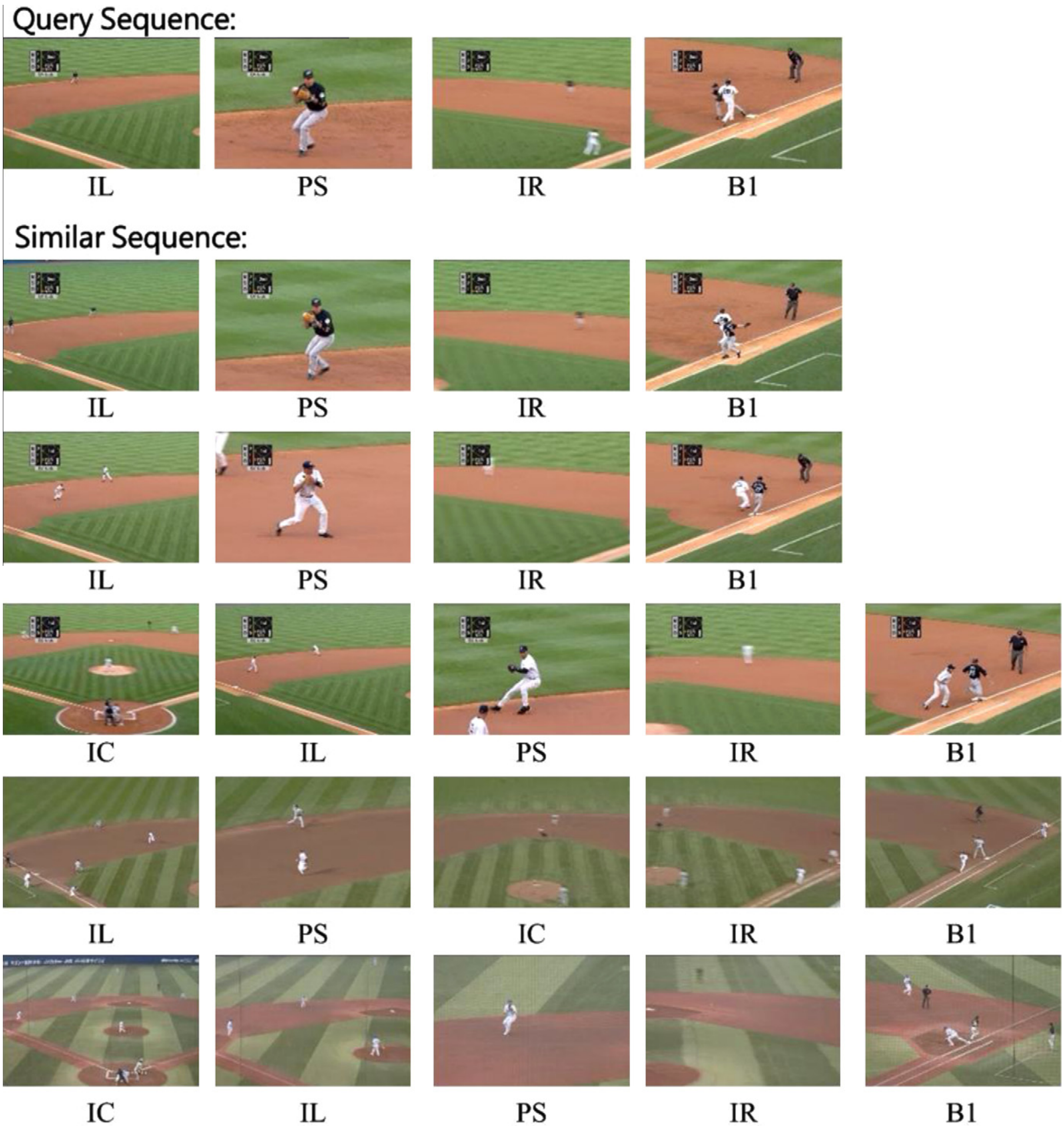

Fig. 17. Example of similar ball hitting event retrieval.

the frames of a ball hitting clip into 16 categories. Thus, each ball hitting event can be represented as a sequence of play region labels (one label per frame). Then, we can apply the dynamic programming algorithm of string-edit distance [32] to measure the distance (dissimilarity) between ball hitting events. The distance between two strings is defined as the minimum number of edit operations. The edit operations include:

Insertion: IL B3 IR B1 → IL B3 PS IR B1.
Deletion: IC IL B3 IR B1 → IL B3 IR B1.
Substitution: IC IL B3 IR B1 → IC IL PS IR B1.

Finally, similar sequences are listed according to their distances to the query sequence in ascending order, together with storyboards and play region strings. An example of similar batting event retrieval is shown in Fig. 17.

To evaluate the effectiveness of the proposed similar event retrieval approach, we use 40 randomly selected query sequences and calculate the average precision for top-k returned similar ball hitting events. Here, we select $k = 1, 3, 5$, and 10. Fig. 18 illustrates the experimental results, where the horizontal axis indicates k and the vertical axis gives the precision. We can see that the precisions of top-3 and top-5 retrieved results are about 85% and 80%, respectively. Even

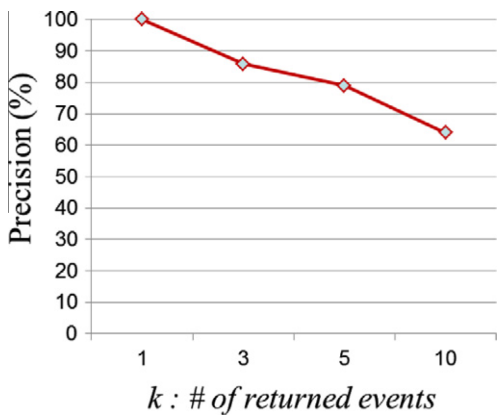

Fig. 18. Precision of top-*k* retrieved results.

though the precision goes down to 64% for top-10 returned results, we still can say that the application of similar event retrieval indeed assists baseball fans and professionals in retrieving, viewing, and comparing similar ball hitting events from different games.

8. Conclusions and future work

In this paper, we propose a HMM-based ball hitting event exploration system for broadcast baseball video capable of spatial pattern detection, frame type classification and event recognition. Convincing results and encouraging performance are obtained. Furthermore, the proposed system also facilitates extensive applications, such as highlight clip extraction by user-designated query, storyboard construction and similar event retrieval.

Compared with existing works on baseball video analysis, the proposed system has some outstanding points. Based on the baseball domain knowledge, we utilize the well-defined field layout and the game-specific spatial patterns to extract more explicit information within a field shot via frame classification, instead of the shot classification which most of the existing works execute. Up to 10 spatial patterns, 16 frame types, and 11 ball hitting events are analyzed and recognized to enhance the robustness and practicability of our system. Extensive applications can be developed based on our proposed spatial pattern detection, frame type classification and event recognition.

There are some limitations in our scheme that open the doors for new exploration. First is that the spatial pattern *B2* is easily missed or mis-detected, which might causes errors in the subsequent processing of frame type classification and event recognition. Since *B2* is typically located on the vertical bisector of the field, it is a possible solution to utilize the symmetry of field layout to assist in detecting and identifying *B2*. Besides, we will apply the proposed system to the video sequences of higher resolution, in which spatial patterns are clearer, and it can be expected that higher accuracy will be achieved. Second, our proposed system works well on MLB and JPB games with prototypical field/stadium layouts. However, some baseball fields/stadiums have different layouts. For example, Koushien baseball stadium, which is one of the most famous baseball stadiums in Japan, has no grass in the infield. Our proposed method may not be able to detect the pitch mound (*PM*) well for this type of baseball stadiums. We have ongoing research for more robust spatial pattern recognition so as to adapt our system to more kinds of baseball video sources. The last limitation is that the proposed system cannot distinguish two different events of similar play region transitions. Possible solutions includes: (1) utilizing scoreboard information to solve some ambiguities of baseball events, (2) locating players and tracking the ball to raise the accuracy of event classification. Furthermore, we will integrate the proposed baseball exploration system in this paper with the research efforts of pitching style recognition [16], baseball trajectory extraction [14], and strike zone shaping [33]. A powerful, practical and professional baseball analysis system will be implemented for automatic content abstraction, pitch-bat strategy inference, and statistics gathering.

Acknowledgments

This work is supported in part by "Aim for the Top University Plan" of the National Chiao Tung University and Ministry of Education, Taiwan, R.O.C., and in part by National Science Council of R.O.C. under the Grant Nos. 98-2221-E-009-091-MY3 and 101-2218-E-009-004-.

References

[1] J. Wang, C. Xu, E. Chng, H. Lu, Q. Tian, Automatic composition of broadcast sports video, Multimedia Systems 14 (4) (2008) 179–193.
[2] X. Yu, H.W. Leong, C. Xu, Q. Tian, Trajectory-based ball detection and tracking in broadcast soccer video, IEEE Transactions Multimedia 8 (6) (2006) 1164–1178.
[3] G. Zhu, C. Xu, Q. Huang, Y. Rui, S. Jiang, W. Gao, H. Yao, Event tactic analysis based on broadcast sports video, IEEE Transactions Multimedia 11 (1) (2009) 49–67.
[4] X. Yu, N. Jiang, L.F. Cheong, H.W. Leong, X. Yan, Automatic camera calibration of broadcast tennis video with applications to 3D virtual content insertion and ball detection and tracking, Computer Vision and Image Understanding 113 (5) (2009) 643–652.
[5] G. Zhu, C. Xu, Q. Huang, W. Gao, L. Xing, Player action recognition in broadcast tennis video with applications to semantic analysis of sports game, in: Proceedings 14th annual ACM International Conference on Multimedia, 2006, pp. 431–440.
[6] E. Kijak, G. Gravier, L. Oisel, P. Gros, Audiovisual integration for tennis broadcast structuring, Multimedia Tools and Applications 30 (3) (2006) 289–311.
[7] 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, Journal of Visual Communication and Image Representation 20 (2009) 204–216.
[8] M.-C. Hu, M.-H. Chang, J.-L. Wu, L. Chi, Robust camera calibration and player tracking in broadcast basketball video, IEEE Transactions on Multimedia 13 (2) (2011) 266–279.
[9] S. Liu, M. Xu, H. Yi, L.-T. Chia, D. Rajan, Multimodal semantic analysis and annotation for basketball video, EURASIP Journal on Applied Signal Processing (2006) 1–13.
[10] H.-T. Chen, H.-S. Chen, S.-Y. Lee, Physics-based ball tracking in volleyball videos with its applications to set type recognition and action detection, in: Proceedings of IEEE International Conference on Acoustics, Speech and, Signal Processing, 2007, pp. I-1097–I-1100).
[11] M. Kumano, Y. Ariki, K. Tsukada, S. Hamaguchi, H. Kiyose, Automatic extraction of PC scenes based on feature mining for a real time delivery system of baseball highlight scenes, in: Proceedings of IEEE International Conference on Multimedia and Expo, 2004, pp. 277–280.
[12] L.-Y. Duan, M. Xu, Q. Tian, A unified framework for semantic shot classification in sports video, IEEE Transactions on Multimedia 7 (2005) 1066–1083.
[13] H. Shum, T. Komura, Tracking the translational and rotational movement of the ball using high-speed camera movies, in: Proceedings of IEEE International Conference on Image Processing, 2005, pp. 1084–1087.
[14] H.-T. Chen, H.-S. Chen, M.-H. Hsiao, Y.-W. Chen, S.-Y. Lee, A trajectory-based ball tracking framework with enrichment for broadcast baseball videos, in: Proceedings of International Computer Symposium, 2006, pp. 1145–1150.
[15] A. Gueziec, Tracking pitches for broadcast television, Computer 35 (2002) 38–43.
[16] H.-T. Chen, C.-L. Chou, W.-J. Tsai, S.-Y. Lee, J.-Y. Yu, Extraction and representation of human body for pitching style recognition in broadcast baseball video, in: Proceedings of IEEE International Conference on Multimedia Expo, 2011.
[17] M.-H. Hung, C.-H. Hsieh, Event detection of broadcast baseball videos, IEEE Transactions on Circuits and Systems for Video Technology 4 (2008) 3829–3832.
[18] W.-T. Chu, J.-L. Wu, Explicit semantic events detection and development of realistic applications for broadcast baseball videos, Multimedia Tools and Applications 38 (1) (2007) 27–50.
[19] Y. Gong, M. Han, W. Hua, W. Xu, Maximum entropy model-based baseball highlight detection and classification, Computer Vision and Image Understanding 96 (2004) 181–199.
[20] M. Fleischman, B. Roy, D. Roy, Temporal feature induction for baseball highlight classification, in: Proc. ACM Multimedia Conference, 2007, pp. 333–336.

[21] C.-C. Lien, C.-L. Chiang, C.-H. Lee, Scene-based event detection for baseball videos, Journal of Visual Communication and Image Representation 18 (2007) 1–14.
[22] C.-C. Cheng, C.-T. Hsu, Fusion of audio and motion information on HMM-based highlight extraction for baseball games, IEEE Transactions on Multimedia 8 (2006) 585–599.
[23] T. Mochizuki, M. Tadenuma, N. Yagi, Baseball video indexing using patternization of scenes and hidden Markov model, in: Proc. IEEE International Conference on Image Processing, 2005, pp. 1212-1215.
[24] P. Chang, M. Han, Y. Gong, Extract highlights from baseball game video with hidden Markov models, in: Proc. of the IEEE International Conference on Image Processing, 2002, pp. 609–612.
[25] A. Hanjalic, Shot-boundary detection: unraveled and resolved?, IEEE Transactions on Circuits and Systems for Video Technology 12 (2) (2002) 90–105
[26] D. Farin, S. Krabbe, P.H.N. de With, W. Effelsberg, Robust camera calibration for sport videos using court models, SPIE Storage and Retrieval Methods and Applications for Multimedia 5307 (2004) 80–91.
[27] D. Farin, J. Han, P.H.N. de With, Fast camera calibration for the analysis of sport sequences, in: Proc. IEEE International Conference on Multimedia and Expo, 2005, pp. 482–485.
[28] B. Jähne, Digital Image Processing, Springer Verlag, 2002.
[29] R.C. Nelson, Finding line segments by stick growing, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (1994) 519–523.
[30] L.R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE 77 (2) (1989) 257–286.
[31] R. Rabiner, B.H. Juang, An introduction to hidden Markov models, IEEE Signal Processing Magazine 3 (1) (1986) 4–16.
[32] E.S. Ristad, P.N. Yianilos, Learning string-edit distance, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (5) (1998) 522–532.
[33] H.-T. Chen, W.-J. Tsai, S.-Y. Lee, Contour-based strike zone shaping and visualization in broadcast baseball video: providing reference for pitch location positioning and strike/ball judgment, Multimedia Tools and Applications 47 (2) (2010) 239–255.